Posters and Demo

Redefining "What" in Analyses of Who Does What in MOOCs

Alok Baikadi[1,2], Christian D. Schunn[1], Yanjin Long[1,2], Carrie Demmans Epp[1,2]
[1]Learning Research and Development Center
[2]Center for Instructional Development and Distance Education
University of Pittsburgh
{baikadi, schunn, ylong, cdemmans}@pitt.edu

ABSTRACT

To advance our understanding of learning in massive open online courses (MOOCs), we need to understand how learners interact with course resources. Prior explorations of learner interactions with MOOC materials have often described these interactions through stereotypes, which does not account for the full spectrum of potential learner activities. A focus on stereotypes also limits our ability to explore the reasons behind learner behaviors. To overcome these shortcomings, we apply factor analysis to learner activities within four MOOCs to identify emergent behavior factors. The factors support characterizations of learner behaviors as driven heavily by types of learning activities and secondarily by time/topic; regression revealed demographic factors (especially country and gender) associated with these activity and topic preferences. Both factor and regression analyses revealed structural variability in learner activity patterns across MOOCs. The results call for a reconceptualization of how different learning activities within a MOOC are designed to work together.

Keywords

MOOCs; learning analytics; online learning; factor analysis

1. INTRODUCTION

With the increasing popularity of massive open online courses (MOOCs), the need to investigate the relationships among learner characteristics, learner-selected activities, and learning outcomes has become critical. Determining these relationships can help us understand how people learn within MOOCs and inform MOOC design and pedagogy. Prior work identified different learning-activity patterns [1, 3] and investigated the relationship between certain types of learning activities and outcomes [7]. Many of these studies were conducted in the context of a single domain or MOOC (e.g., [1]). Furthermore, little work has investigated how demographic variability could lead to different behavioral patterns in MOOCs, leaving an open question: Can the identified patterns be generalized across instructional domains and populations?

Until recently, studies of learning within MOOCs focused more on the number of learners being served than pedagogy [6]. This focus on their size has left many facets of MOOCs underexplored and poorly understood [1]. These aspects include a need to understand how learners engage with MOOCs [1], their behavior patterns, and their motivations [3]. Understanding these factors may allow us to design courses that support the learning activities and outcomes that learners want.

We investigate learning patterns in four MOOCs based on learner activities across courses from different disciplines. We used the activity-centered data reduction technique of factor analysis to identify the underlying course activities that describe learner activity patterns within each offering of the selected MOOCs. The factor analyses applied to 10 MOOC offerings enabled us to identify 1) factors that are common to most of these MOOCs and 2) factors that are less common. Regression analyses were then used to examine the relationship between learner demographic variables and their participation on each factor. These analyses support the distinctions between factors and the presence of varied factors across MOOCs.

This investigation is among the first to identify and compare activity patterns and demographic influences across learning domains. The results improve our understanding of learner behaviors across contexts and could inform the design of more domain-sensitive learning experiences.

2. LEARNER ACTIVITIES IN MOOCS

Research into MOOCs has spanned a range of topics, with recent discussions becoming more nuanced. Work that has investigated how learners interact with a MOOC [5] found that their behaviors can be characterized through a set of trajectories rather than the commonly used completion and attrition model. These trajectories through graded assignments and lecture videos within computer science MOOCs characterize how different types of learners used some of the course materials to support their learning activities [1]. The identified usage patterns included those who mostly watched lectures, mostly submitted assignments, performed some combination of these activities, downloaded course resources, or registered but did very little.

Some researchers have taken the next step by linking these types of activities (watching video lectures, submitting assignments, and discussion forum activity, types of questions asked) to course performance (certificate earned, learning outcomes and gains, course completion) [2, 7]. To obtain a better understanding of how these and other factors influence learner success within MOOCs, the relationships among socio-demographic variables, student activities, and learner success have been explored. The most common predictors of certificate earning and completion were prior education [2], sex [4], and country of origin [4].

3. MOOC CORPUS

Data from the 132,324 learners who performed at least one action (taking a quiz, posting to the forum, or watching a video) in 4 of the University of Pittsburgh's Coursera MOOCs were used. To describe learner activities within a range of course types and explore generalizability across disciplines, courses from different domains were chosen: health sciences (nutrition for health and clinical terminology), education (accountable talk), and public health (disaster preparedness). Data from multiple offerings (Jan. 2013 – Dec. 2015) of the same course were used when available.

The courses lasted 6 or 7 weeks. The core materials for each week consisted of video lectures and a quiz. Some weeks included assignments, disaster preparedness used peer-assessment, and accountable talk had a project. Clinical terminology incorporated multimedia modules that enabled the learner to interact with learning resources. Since these modules presented core content, they were labeled as lectures. Only the Clinical Terminology instructors explicitly encouraged discussion forum use and provided study tips. This variability provided a cross-section of course formats that enables us to identify learner activities that apply across courses and that are specific to a course. We used the activity counts for each forum, quiz, and lecture video.

4. RESULTS

4.1 Learner Activities

Factor analysis with varimax rotation was used to reduce the dimensionality of the data and identify learners' underlying behavioral tendencies. Course activities that at least 1% of active learners performed were used. To test the stability of the patterns, a separate factor analysis was conducted for each course offering. Factors that accounted for at least 5% of the variance were kept.

In 3 of the 4 courses, activities were largely grouped into 4 factors: lecture activity, quiz activity, forum participation and participation in activities from weeks 1 and 2. In contrast, clinical terminology shows more depth in weekly content: lecture activity is represented by 4 factors, each capturing a 1-2 week span. For quizzes, we see three factors: summative quizzes presented at the end of each module, early quiz activities, and later quiz activities.

4.2 Predicting Activities Using Demographics

We calculated a factor score for each learner, which indicates a tendency towards the behavior described by that factor. For example, a learner with a high score for the lectures factor would have viewed more lectures than one with a low score. A general linear model (GLM) was used to predict learner factor scores from learners' socio-demographic characteristics. Only those (n = 2963) with individual demographic profiles were included. We applied GLM to courses that had contrastive factor structures: the second offering of nutrition for health represented those with media-based factors and the first offering of clinical terminology represented those with time-based factors.

For clinical terminology, we aggregated early lecture factors, late lecture factors, and quiz factors to create factors that were comparable to the other courses. We then ran a generalized linear model predicting each of these aggregated factors.

Each factor is influenced differently by learner demographics and are contrasted between the two courses. For example, the early lecture watching factor from nutrition for health was more strongly associated with female learners than males. This was not the case for clinical terminology. Late lecture watching activity was predicted by learner age for both courses. However, a difference in factor scores for the younger and older populations for those in the middle age groupings is visible between the courses. Within clinical terminology, we also see that some age groups are more active earlier in the course than later. Additional differences in how demographic variables predict factors are visible when considering learners' quiz participation and their continent of residence. Similar factor scores are seen for those who live in Asia and North America when considering learner activities within clinical terminology. This similarity does not hold across courses; learners from Asia and Europe appear to be more similar in their quiz taking habits when considering the data from nutrition for health.

5. CONCLUSION

Our factor and regression analyses across multiple offerings of the same course show that learner behaviors are relatively consistent across time. However, differences in factors across courses suggest that design and domain affect how learners select learning content and activities, which requires further study.

Our work is among the first applications of exploratory factor analyses across learner activities within MOOCs from different domains. Prior work has focused on a person-based approach that describes the behavior patterns of individuals by assigning them to canonical groups. This work, therefore, provides a new lens to examine the full range of learner behaviors in MOOCs.

6. REFERENCES

[1] Anderson, A., Huttenlocher, D., Kleinberg, J., and Leskovec, J. 2014. Engaging with Massive Online Courses. In *Proceedings of the 23rd International Conference on World Wide Web*, ACM, 687–698. DOI = http://doi.org/10.1145/2566486.2568042

[2] Breslow, L. B., Pritchard, D.E., DeBoer, J., Stump, G.S., Ho, A.D., and Seaton, D. T. 2013. Studying learning in the worldwide classroom: Research into edX's first MOOC. *Research & Practice in Assessment* 8: 13–25.

[3] Gasevic, D., Kovanovic, V., Joksimovic, S., and Siemens, G. 2014. Where is research on massive open online courses headed? A data analysis of the MOOC Research Initiative. *The International Review of Research in Open and Distributed Learning* 15, 5.

[4] Kizilcec, R.F. and Halawa, S. 2015. Attrition and Achievement Gaps in Online Learning. In *Proceedings of the 2nd ACM Conference on Learning @ Scale*, ACM, 57–66. DOI = http://doi.org/10.1145/2724660.2724680

[5] Kizilcec, R.F., Piech, C., and Schneider, E. 2013. Deconstructing Disengagement: Analyzing Learner Subpopulations in Massive Open Online Courses. In *Proceedings of the 3rd International Conference on Learning Analytics and Knowledge (LAK)*, ACM, 170–179. DOI = http://doi.org/10.1145/2460296.2460330

[6] Kovanović, V., Gašević, D., Joksimović, S., Siemens, G., and Hatala, M. 2015. MOOCs in the News: A European Perspective. In *Proceedings of "WOW! Europe embraces MOOCs."*

[7] Wang, X., Yang, D., Wen, M., Koedinger, K., Rosé, C. P. (2015). Investigating how student's cognitive behavior in MOOC discussion forums affect learning gains. In *Proceedings of the 8th International Conference on Educational Data Mining (EDM)*.

Text Classification of Student Self-Explanations in College Physics Questions

Sameer Bhatnagar
Polytechnique Montreal

Michel Desmarais
Polytechnique Montreal

Nathaniel Lasry
John Abbott College

Elizabeth S. Charles
Dawson College

ABSTRACT

This study looks at the text data generated from the Asynchronous Peer Instruction tool, DALITE. The goals of this work are two-fold: i) to determine whether the words students use in their self-explanations can be predictive of their success on the related multiple-choice item, or even reveal their uncertainty about the concept being tested; and, ii) to determine if the collection of words used by a student over the course of a semester using DALITE can predict their end-of-semester learning outcomes. Through the course of this study, we examine the effectiveness of different statistical models and document representations to explain these data. Weak results suggest richer syntactic ans semantic models of text are needed.

1. INTRODUCTION

The Distributed Active Learning Integrated Technology Environment (DALITE)[2], implements an original peer instruction paradigm that relies on students providing a rationale to their choice over multiple-choice questions (MCQ). After every MCQ, the student is prompted to provide the rationale for their choice. Once provided, the student is shown a few other students' rationales for the same choice, and for an alternate choice. If the answer was right, the alternate choice shown is for a wrong answer, else it is the right answer's rationales. The student can then decide to change their choice or not. This instruction paradigm has recently been integrated into the EdX platform and we believe it has a great future in MOOCs and other environments where educational crowdsourcing bootstraps instructional content. However, for the bootstrap to be effective, a good understanding of the process of learning from this type of content is crucial. This paper reports on early analysis of student rationales with this aim in mind, using a text classification framework. For this particular study, we are interested in

- identifying students who are unsure about their answers (as revealed by when they switch from right-to-wrong, or wrong-to-right in DALITE). Are there linguistic patterns for students who are uncertain?
- studying the effect of the teacher on the development of their students' language. Is there a teacher effect?
- documenting group differences in language use, for subpopulations such as strong vs. at risk students, or male vs. female. [6] discusses the gender gap in performance in college physics classrooms. This was observed in a previous study of ours looking at DALITE as well[1]. Is there a measurable difference between the language used by strong students and weak ones? Are there gender differences?
- finding minimally disruptive, low-stakes, language based predictors of student failure, as early in the semester as possible. Can the results of DALITE questions assigned prior to any of the three midterms predict which students ultimately fail?
- which classification algorithms perform the best in this context? What document representations optimize classifier performance for the different target variables?

2. DATA AND METHODS

2.1 Corpus Statistics

The dataset is made up of student-generated self-explanations for 80 different DALITE items (conceptual physics questions). On average, 97 students attempted each item, writing explanations for each question with an approximate length of 32 words, with a type-token ratio of 0.87. The average number of unique words used by all students to answer any given one item was 310. The 140 students in this study came from three different colleges in the province of Quebec, Canada. The course material was surrounding what would normally be freshman physics in the U.S. Besides collecting midterm grades and final course grades, each student also completed the Force Concept Inventory[4], at the beginning of the term, as well at the end. The normalized pre-post gain (or Hake gain) on this questionnaire has become a standard measure in the physics education research community. More aggregate statistics of the dataset rest are more fully described in [1].

2.2 Statistical Models

Significant amount of work was done in comparing different statistical learning algorithms for text classification. One of the simplest yet most effective text classification approaches

is the Naive Bayes classifier[7]. In datasets when vocabulary size was small, [8] compared different event models for the Naive Bayes family of classifiers, finding that the multivariate Bernoulli model (where the components of each document vector are binary, modeling simply the presence or absence of a word), performed better for text classification than its multinomial counterpart (where document vectors are the counts of the different terms in that document). [5] shows that Support Vector Machines (SVM) are well suited to the task of text classification, due to three factors inherent to the nature of the task: high dimensional feature space, many relevant features (dense concept vectors), but sparse document vectors. Finally, we explore the utility of a k-nearest neighbor classifier in this setting as well, based on the intuition that the document vectors might not be linearly separable.

2.3 Document Vector Representations

This study also aims to explore different choices of document representation. The most basic choice would have the elements of document vectors simply containing raw word counts (we ensure that the words in the original questions item text are always included in the term-document matrices).[9] showed that shifting importance to rarer words across a corpus would improve classifier effectiveness. We also look at N-grams to relax the independence assumption between words, but this may require more data than we have to avoid sparsity (we only go up to bigrams). There is an interest in also adding syntactic information, such as part-of-speech (POS) tags, and represent documents as bags of POS-tags (e.g. since there is an important difference in physics between using the word "force" as a verb or as a noun, which could reveal a misconception if students use it incorrectly). Finally, document vectors can also be represented for their semantic content. One of the most successful techniques for this is Latent Semantic Analysis[3], which relies on a truncated singular value decomposition of term co-occurrence matrices. This allows us to approximately represent documents in a lower dimensional space, and typically removes noise such that document vectors that are similar in meaning, cluster together. The sensitive choice in such latent factor models is the choice of how many factors will be kept after the matrix decomposition. We do a grid search over different possible number of dimensions to reduce to, ranging from 2 to 10, and pick the model that performs best in cross-validation.

3. DISCUSSION

None of the results are presented here, due to space limitations.[1].Our research team started this study with the following question: do students in different cognitive states, use different words to explain their thinking when answering conceptual questions? In general, the poor performance of most of the statistical models studied herein tends to confirm the intuition behind the body of work centered around Latent Semantic Analysis: in most cases, the mere occurrences of the words is not enough to discriminate strong students from weak ones, and that such datasets can be too noisy and sparse. The inability of all these models to predict item-level outcomes, such as getting the answer correct, or whether a student is about to switch their answer, leads us to believe that richer syntactical and semantic representations will be required.

[1]All scripts used to get the results, for this study are available at sameerbhatnagar.github.io/

4. FUTURE WORK

The most important facet of DALITE that has not yet been studied lies in the patterns in student preferences: when students are on the page where they can read their peers' rationales, and are asked to reconsider their original answer choice, they are also prompted to *select which, if any, of their peers' rationales they thought was most convincing.* This 'crowdsourcing' of high quality, peer-assesed rationales if very healthy for the future of DALITE, but is also fertile ground for research related to the current study: what distinguishes language that is effective to convincing to students (whether for the right answer, or the wrong one)?

5. ACKNOWLEDGMENTS

We would like to thank the teachers who participated in this study, without whose support this work would not be possible: Chris Whittaker (Dawson College), Kevin Lenton (John Abbott College), and Kevin Lenton (Vanier College). This work was funded through *Programme de Recherche sur l'Apprentissage et l'Enseignement* from the government of Quebec. Finally, we remain indebted to all our students who actively contributed to DALITE through their participation.

6. REFERENCES

[1] S. Bhatnagar, M. Desmarais, C. Whittaker, N. Lasry, M. Dugdale, and E. S. Charles. An analysis of peer-submitted and peer-reviewed answer rationales, in an asynchronous peer instruction based learning environment.

[2] E. Charles-Woods, C. Whittaker, M. Dugdale, N. Lasry, K. Lenton, and S. Bhatnagar. Designing of dalite: Bringing peer instruction on-line. In N. Rummel, M. Kapur, M. Nathan, and S. Puntambekar, editors, *Computer Supported Collaborative Learning.*

[3] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990.

[4] D. Hestenes, M. Wells, and G. Swackhamer. Force concept inventory. *The physics teacher*, 30(3):141–158, 1992.

[5] T. Joachims. *Text categorization with support vector machines: Learning with many relevant features.* Springer, 1998.

[6] L. E. Kost, S. J. Pollock, and N. D. Finkelstein. Characterizing the gender gap in introductory physics. *Physical Review Special Topics-Physics Education Research*, 5(1):010101, 2009.

[7] C. D. Manning, P. Raghavan, H. Schütze, et al. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[8] A. McCallum, K. Nigam, et al. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Citeseer, 1998.

[9] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

Automated Feedback on Group Processes: An Experience Report

Marcela Borge
Pennsylvania State University
301C Keller Building
University Park, PA 16802
mborge@psu.edu

Carolyn P. Rosé
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
cprose@cs.cmu.edu

ABSTRACT
We report on an effort to evaluate the efficacy of automated assessment and feedback of the quality of collaborative discourse in the context of an online project based course. Results of automated assessment and impact on collaborative process are evaluated over a semester-long course.

Keywords
Collaborative learning, automated process assessment

1. INTRODUCTION
In this paper we report on an effort to evaluate the efficacy of automated assessment and feedback of group processes in the context of an online project based course. It is well known that the positive effects of collaborative learning are not guaranteed. Instead, those benefits depend upon the quality of collaborative interactions that occur during activity [1]. This is problematic since most students lack the cognitive skills necessary to engage in high quality collaborative interactions [3]. Research suggests that developing socio-metacognitive expertise, the ability to understand, monitor, and regulate collective thinking processes that occur during collaboration, can help to mitigate group dysfunction and optimize collaborative interactions [4].

We have been working on developing activity design models to inform the design of Computer Supported Collaborative Learning (CSCL) systems to support socio-metacognitive development [4]. In this paper, we describe an approach to automated, collaborative discourse assessment and a study we ran in a real educational environment. We focus on two areas of inquiry motivated by emerging research. First, (RQ1) How reliably can we automatically assess collaborative discussion quality and (RQ2) does automated assessment impact future performance differently than human generated feedback?

2. METHODS
2.1 Study Context
The study took place during a 16-week, introductory, undergraduate, online course on information sciences and technology. Fourty-one online students participated in the study, each belonging to one of 14 groups. As part of the course, students were required to read a chapter from the textbook or supplementary materials each week. Students were assigned to teams within the first four weeks of the semester. Then, in weeks five, seven, nine, eleven, and fourteen, students participated in a synchronous discussion related to the reading materials. The discussion sessions were held in a collaborative workspace with chat capabilities called CREATE.

2.2 Research Design
Across the five time-points during which students engaged in a collaborative chat activity, we compared the effect of four different feedback conditions on the quality of collaboration at the next time point. After each of the first four discussion tasks, groups were assigned to one of four feedback conditions that determined the type of feedback they received at that time point.

The study was run as a within-subject manipulation. The four conditions included: (1) no feedback, (2) expert feedback, (3) automated feedback, and (4) best practices. Those in condition one received no feedback about the quality of their processes. Those in condition two received feedback from trained research assistant who would analyze their processes using our coding construct. Condition three received feedback based on automated assessment of processes. Condition four was given feedback based on common strengths and weaknesses of collaborative groups [4] and not based on the group's specific processes. All feedback was worded in a consistent manner such that teams would not know what condition they received.

An assessment of group processes was conducted for each discussion based on the transcripts from the chat environment that housed the activity. Team process measures at the first time point were used to identify groups' initial strengths and weaknesses. Thus, the first assessment was treated as a baseline, and each subsequent measurement, controlling for the previous assessment, was treated as a measure of the effectiveness of the form of feedback experienced after the previous discussion.

2.3 Assessment of Collaborative Discourse Quality
After each discussion session, individual students completed an evaluation of the quality information synthesis and knowledge negotiation in their group.

In the assessment rubric, there are three categories of behavior within each of the two core capacities, with each category assessed on a five-item, ordinal scale. The first core capacity, information synthesis, consists of three categories of discourse behavior: verbal participation, developing joint understanding, and joint idea building. Verbal participation examines the amount of turns of speech contributed by each member relative to the team's total turns of speech. Developing joint understanding evaluates the extent to which teams make an effort to ensure that members fully understand the ideas presented by taking time to reword, rephrase, or ask for further clarification of shared information. Joint idea building focuses on the extent to which team members elaborate on another member's contribution in

order to ensure that information introduced by any member is not ignored or accepted, without discussion.

The second core capacity, knowledge negotiation, also consists of three categories of behavior. These categories are contributing alternative ideas, quality of claims, and norms of evaluation. Contributing alternative ideas evaluates the extent to which teams present and discuss alternative perspectives, claims, or suggestions. Quality of claims focuses on evaluating the extent to which teams provide logical, fact-based evidence and rationale. Norms of evaluation focuses on evaluating the extent to which teams adhere to social norms that promote the development of psychological safety.

Twenty percent of the total data was double coded by the research assistant and another trained graduate student to determine inter-rater reliability of the instrument: $r = 0.86$; $p < 0.001$, Kappa = 0.64; $p < 0.001$. Once each item of a core capacity is rated, they are averaged to produce a single Collaborative Discussion Quality score, which is a continuous value between 0 and 5 that we use to track improvement over time in collaborative discussion processes in the analysis below.

2.4 Automated Assessment

A key component of the study is an evaluation of an automated assessment technique. The six scales that comprise the three dimensions of each of the two core competencies in the assessment rubric were automatically predicted based on distributions of automatically predicted process codes. Training data for the macro level regression model for the 6 scales was a corpus of 13 discussions (with a total of 7015 turns) that were hand coded with a process-analysis coding scheme developed as part of this work. We built on a coding scheme developed for a laboratory study [3], but modified it for use in a real-world classroom setting. Each discussion was hand coded at the turn level using the process analysis and then assessed along the 6 different dimensions. We established inter-rater reliability for this schema of Kappa= .74, indicating substantial reliability.

The automated process analysis models were trained using the LightSIDE tool bench. We extracted a feature space consisting of unigrams, bigrams, POS bigrams, and a line length feature, and used a Logistic regression classifier with L2 regularization to avoid over-fitting. In a leave-one-team-out cross-validation, we achieved an accuracy of 86% and kappa of .77. The assessment needed in order to generate feedback for the study is at the level of the six scales that rate two core competencies, with three dimensions each. We used the counts of predicted process codes per team to predict these six scales using a separate linear function trained using a simple linear regression for each scale.

We expected a drop in performance when applying a model trained in a previous experiment. In the initial week of the study, we used the model trained on the earlier data to generate the six scores per team. In subsequent weeks of the study, we retrained the simple linear regression models to predict hand coded assessment scores from data collected in the current study during the earlier weeks of the semester. The process coding that created the predictor variables for those regression equations was computed using the original trained process coding models.

3. RESULTS

At each of four time points in the course, we collected automated assessments of collaborative process in terms of the six assessment dimensions. Each time, each of three to four groups was assigned a rating on a 5-point scale for each of the six dimensions. The same assessments were also made by human raters in order to assess the quality of the automated rating. Over time, we continued to use the original turn level process models but adapted the simple linear regressions to compute the six scale measures from the counts of the turn level codes using the hand rated data collected in the second course so far. We evaluate the quality of the automated rating by computing a kappa with linear weighting between the sets of automated ratings and human ratings. At time point one, before any data from the second instance of the course was available, the automated ratings were assessed to be at random. By time point two, the weighted kappa was .19. It was better at time point three, specifically .4. And finally, at time point four, it was up to .58. Altogether ratings for 10 sessions of the second course were needed to adapt the models and achieve a weighted kappa of .58.

Given that the automated feedback generated at early time points in the course was based on poor quality assessments, an important question is how much of a negative impact these errors cause for students. We measured the effect of the experimental manipulation using a repeated measures ANCOVA for each scale assessment separately. In each case, the dependent measure was the scale assessment at a time point rated by an expert rater, the covariate being that scale assessment at the previous time point, the independent variable being the condition that generated the feedback received by the team at the previous time point, and time point as a nominal control variable. We did not observe any consistent improvement over time or significant effect of condition on any one of the six scale assessments.

4. CONCLUSIONS

In this paper we addressed important questions related to the automated assessment of collaborative discourse quality in real educational settings. Though the automated process analysis was evaluated as very reliable within the course that provided the training data, the automated assessments in the second run of the course were initially very poor and only improved after 3 weeks of data were collected to use for adapting the prediction models.

5. ACKNOWLEDGMENTS

This work was funded by NSF grant IIS-1320064.

6. REFERENCES

[1] Barron, B. (2003). When smart groups fail. *Journal of the Learning Sciences*, 12(3), 307-359.

[2] Biber, D. & Conrad, S. (2011). *Register, Genre, and Style*. Cambridge University Press.

[3] Borge, M., & Carroll, J. M. (2014). Verbal Equity, Cognitive Specialization, and Performance. In *Proceedings of the 18th International Conference on Supporting Group Work*, 215–225.

[4] Borge, M., Ong Shiou, Y., & Rosé, C. 2015. Design models to Support the Development of High Quality Collaborative Reasoning in Online Settings. In *the Proceedings of the International Conference of Computer Supported Collaborative Learning (CSCL) 2015,* Volume 2, 427-434.

Mining Sequences of Gameplay for Embedded Assessment in Collaborative Learning

Philip Buffum
North Carolina State University
Computer Science
psbuffum@ncsu.edu

Megan Frankosky
North Carolina State University
Psychology
rmhardy@ncsu.edu

Kristy Elizabeth Boyer
University of Florida
Computer & Information Science & Engineering
keboyer@ufl.edu

Eric Wiebe
North Carolina State University
STEM Education
wiebe@ncsu.edu

Bradford Mott
North Carolina State University
Computer Science
bwmott@ncsu.edu

James Lester
North Carolina State University
Computer Science
lester@ncsu.edu

ABSTRACT
This poster presents a sequence mining analysis of collaborative game-based learning for middle school computer science. Using pre-post test results, dyads were categorized into three groups based on learning gains. We then built first-order Markov models for the gameplay sequences. The models perform well for embedded assessment, classifying gameplay sequences with 95% accuracy according to whether the group learned the target concepts or not. These results lay the groundwork for accurate embedded assessment of dyads in game-based learning.

Keywords
Embedded assessment; game-based learning; collaboration; Markov models

1. INTRODUCTION
There is growing recognition of the importance of collaborative learning, in which students work together to solve problems [2, 3]. Collaboration, furthermore, can have an especially beneficial impact in game-based learning, where it has been shown to promote significant student learning gains [4] and provide significant motivational benefits [8], as well as deliver more equitable gaming experiences for diverse learners [1, 6].

Yet collaborative learning presents unique challenges to educational data mining research. While much current work in this field relies on mapping individual students' outputs, student collaboration produces learning that plays out as a joint activity, necessitating different approaches to understanding the underlying processes [7]. Recent work in educational data mining has demonstrated some success in predicting student outcomes in paired learning, as long as both students in the pair have similar initial knowledge [5].

This poster examines collaborative game-based learning in the context of the ENGAGE game-based learning environment, with which middle school students learn about computer science through an overarching narrative situated within a fictional underwater research station. In this study, students played ENGAGE in pairs at a single computer, taking turns with one set of game controls. These two students' inputs were therefore captured within a single gameplay log. The analysis presented here investigates a variation on the traditional learning question of, "Did student *S* learn the concept?" and instead asks, "Did the collaborative partnership *P* result in learning?" By building first-order Markov models on dyads' gameplay logs, we discovered that the gameplay sequences of dyads in which some learning occurred (i.e. at least one of the students learned the material) differed significantly from those in which no learning occurred, and moreover, that we can classify with very high accuracy the learning that occurred on a targeted learning objective.

2. COLLABORATIVE LEARNING TASK
This study focuses on a subset of the ENGAGE game. In ENGAGE'S Digital World level, students learn how computers process data using the binary number system. The current analysis focuses on one room in the game world, in which students integrate the two concepts of *variables* and *binary numbers*, having earlier explored both these individual concepts in isolation from one another. 124 middle school students played the game in pairs; as there is one gameplay trace for each dyad, this produced 62 gameplay traces. We administered individual pre- and post-tests to each student so that we could characterize each student's learning outcomes. The goal of the present analysis is to utilize gameplay logs to predict learning, specifically to investigate how the gameplay of those dyads who scored higher on learning assessments differs from the gameplay of those who did not score higher. Accordingly, having assigned each *individual* student a grade based on pre and post test scores, we then classified student *pairs* into one of three categories: *Learner* (19 dyads), *Prior Mastery* (23 dyads), and *Non-Learners* (20 dyads).

3. RESULTS
The modeling approach aims to identify differences in gameplay sequences between students in the *Learner*, *Prior Mastery*, and *Non-Learner* groups. We began with one of the simplest sequential models of all, first-order observable Markov models. It was expected that more sophisticated models, such as hidden Markov models or Conditional Random Fields, may be needed to characterize the gameplay sequences well; however, as this poster demonstrates, the simplest model was able to classify the gameplay sequences of *Learner*, *Prior Mastery*, and *Non-Learner* groups with high accuracy.

We built separate models for each group (*Learner, Prior Mastery, Non-Learner*) and then determined whether there were significant differences in the models for each group by comparing model fit (in terms of log-likelihood, since the probabilities themselves are very small in magnitude). We performed this pairwise comparison for all three groups, as described below:

1. For each gameplay trace sequence s_i in the Learner group:
   i. Compute $logProb(s_i \mid L_{leave\text{-}i\text{-}out})$ of observing s_i under the Learner model L (trained in a leave-one-out fashion where s_i was the left-out sequence).
   ii. Compute the log-likelihood $logProb(s_i \mid PM)$ of observing s_i under the Prior Mastery model PM trained on all Prior Mastery gameplay sequences.
   iii. Compute the log-likelihood $logProb(s_i \mid NL)$ of observing s_i under the Non-Learner model NL trained on all Non-Learner gameplay sequences.
2. Repeat the analogous process for each gameplay sequence in the Prior Mastery and Non-Learner groups.
3. For each group's sequences, test whether the set of log-likelihoods for that group under its own model is significantly higher than the log-likelihoods for that group under the other groups' models.

The models were significantly different across *Learners*, *Prior Mastery*, and *Non-Learner* groups, as shown in Figure 1, which shows the absolute values of log likelihoods for each of the three categories. In this graph, a **lower** absolute log-likelihood indicates better model fit. For each category, the graph shows three bars, the first showing the log likelihood for the given category's sequences under the *Learner* model, the second bar showing the log likelihood for the given category's sequences under the *Prior Mastery* model, and the third bar showing the log likelihood for given category's sequences under the *Non-Learner* model. We conducted a series of paired *t*-tests to determine, for each group, whether there were significant differences between the log likelihoods for its own model and those for the other two models. For the *Learner* group model, its own log likelihoods were found to be significantly better than the log likelihoods of the other two models at the $p < .01$ level. For both of the other two models, *Prior Mastery* and *Non-Learner*, their own log likelihoods were found to be significantly different than the other respective models with even greater significance, at the $p < .001$ level.

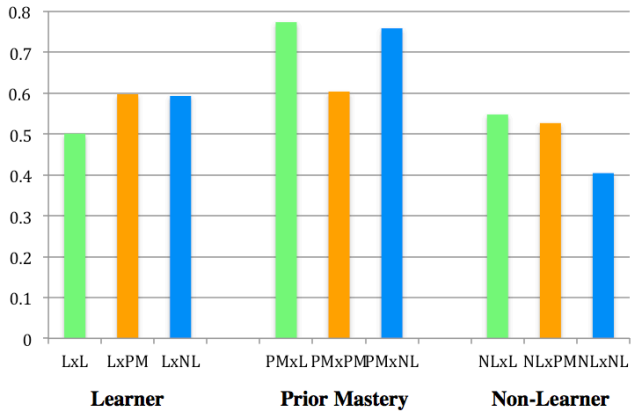


Figure 1. Absolute value of log likelihoods for each of the three categories. Lower values indicate better model fit.

Finally, we investigated the extent to which these models could classify *Learner, Prior Mastery*, and *Non-Learner* based only on the observed gameplay sequences in Room 2 and using leave-one-out cross-validation. A sequence was labeled with the group whose model produced the highest log-likelihood for that sequence (using only models that were trained with the sequence left out). Using this classifier, for the *Learner* category, 89.5% of pairs (17 out of 19) were correctly classified. For the *Prior Mastery* category, 100% of pairs (23 out of 23) were correctly classified. For the *Non-Learner* category, 95% (19 out of 20) were correctly classified. On the whole, this reflects a 95.2% accuracy in classifying whether a collaborative pair of students would be in the *Learner*, *Prior Mastery*, or *Non-Learner* group.

4. CONCLUSION

Modeling collaborative learning is an important direction for educational data mining research. We have demonstrated that sequence modeling relying on first-order Markov models can differentiate gameplay sequences of pairs where at least one partner learned from pairs who did not learn. Moreover, these models can classify those gameplay sequences with very high accuracy according to whether the dyad learned or not.

The opportunities are numerous for empirical studies into collaborative gameplay, problem solving, and dialogue. For example, the current analysis assumes that the maximal knowledge of the group is expressed through gameplay, an assumption that needs to be investigated. Additionally, a natural next step is to examine prediction power of individual learning along with the slightly more abstracted dyadic learning considered here. It is hoped that this line of investigation will move us toward highly effective support of dyadic learning.

5. REFERENCES

[1] Buffum, P.S. et al. 2016. Collaboration and Gender Equity in Game-Based Learning for Middle School Computer Science. *Computing in Science & Engineering*. 18, 2 (Mar. 2016), 18–28.

[2] Coleman, B. and Lang, M. 2012. Collaboration Across the Curriculum: A Disciplined Approach to Developing Team Skills. *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE '12)* (2012), 277–282.

[3] Falkner, K. et al. 2013. Collaborative Learning and Anxiety: A phenomenographic study of collaborative learning activities. *Proceedings of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)* (2013), 227–232.

[4] Hickey, D.T. et al. 2009. Designing Assessments and Assessing Designs in Virtual Educational Environments. *Journal of Science Education and Technology*. 18, 2 (Feb. 2009), 187–208.

[5] Rafferty, A. et al. 2013. Estimating Student Knowledge from Paired Interaction Data. *Proceedings of the 6th International Conference on Educational Data Mining* (2013).

[6] Richard, G.T. and Hoadley, C. 2015. Learning Resilience in the Face of Bias: Online Gaming, Protective Communities and Interest-Driven Digital Learning. *Proceedings of the 11th International Conference on Computer Supported Collaborative Learning* (2015), 451–458.

[7] Stahl, G. et al. 2006. Computer-supported collaborative learning: An historical perspective. *Cambridge handbook of the learning sciences*. (2006), 409–426.

[8] Warren, S.J. et al. 2008. A MUVE Towards PBL Writing. *Journal of Research on Technology in Education*. 41, 1 (Sep. 2008), 113–140.

Can Word Probabilities from LDA be Simply Added up to Represent Documents?

Zhiqiang Cai
University of Memphis
Memphis, TN, USA
zcai@memphis.edu

Haiying Li
Rutgers University
New Brunswick, NJ, USA
haiying.li@gse.rugers.edu

Xiangen Hu
University of Memphis
Memphis, TN, USA
xhu@memphis.edu

Art Graesser
University of Memphis
Memphis, TN, USA
agraesser@memphis.edu

ABSTRACT
This paper provides an alternative way of document representation by treating topic probabilities as a vector representation for words and representing a document as a combination of the word vectors. A comparison on summary data shows that this representation is more effective in document classification.

Keywords
Topic modeling, LDA, document clustering, cluster similarity

1. INTRODUCTION
Topic modeling has been one of the most important methods in natural language analysis. It helps to discover underlying topics in a collection of documents. The found topics are used to form topic features for documents. The topic features are then used as input to perform task such as document clustering [11], automated summarization [1], automated essay grading [6], etc. LDA (Latent Dirichlet Allocation) [2, 3] is the most popular way for topic modeling. LDA topic model provides topic proportions as a vector representation of document. We investigated an alternative way of document representation by summing up word probabilities from LDA topic model. The new representation is compared with the topic proportion representation as input of a document clustering task on a summarization data set. The results showed that the simple "probability sum" document representation performs better.

2. LDA and Document Representations
Latent Dirichlet allocation (LDA), first introduced by Blei, Ng and Jordan in 2003 [3], is one of the most popular methods in topic modeling. LDA represents topics by word probabilities. Given a vocabulary with N words, $\{w_1, w_2, \cdots, w_N\}$, the LDA model probabilities $\boldsymbol{P}_k = (p_k(w_1), p_k(w_2), \cdots, p_k(w_N))$ form a representation of the k^{th} topic $(k = 1,2,\cdots,K)$. The words with highest probabilities in each topic usually give a good idea about what the topic is.

In LDA, a document d has an inferred topic proportion which is usually used as topic features to represent the document:

$$\boldsymbol{T}(d) \sim (t_1(d), t_2(d), \cdots, t_K(d)).$$

From the point of view of statistics, topic proportion is probably the only choice for LDA-based document representation. However, if we jump out of the box of statistics, we can simply view the word probabilities across the K topics as a K -dimensional vector representation for each word. Thus, a document can be represented by summing up the word probability vectors:

$$s_k(d) = \sum_{i=1}^{N} p_k(w_i) \log(1 + f(w_i, d)), (k = 1, 2, \cdots, K)$$

In the above formula, $s_k(d)$ is the "probability sum" of the document d on the k^{th} topic, $p_k(w_i)$ is the probability of the word w_i on the k^{th} topic, and $f(w_i, d)$ is the frequency of the word w_i in the document d. The logarithm of word frequency is known as Zipf scale [9].

3. Corpus for Document Clustering
201 participants wrote 1481 summaries for 8 passages, about 185 for each passage [10]. The lengths of the passages ranged from 195 to 399. The Flesch-Kincaid grade level was from 8.6 to 11.7. Some passages had similar topics: *Working and Running*, *Kobe and Jordan*, and *Effects of Exercising* on sports and exercising; and *Floods* and *Hurricane* on disasters.

The summaries were collected from an online experiment. The original goal was to evaluate the effect of an online AutoTutor [5, 9] lesson that teaches summarization. Each subject composed summaries for 2 texts before learning the lesson, 2 after learning, and 4 during learning with a counter-balanced design. The participant wrote each summary immediately after reading a passage. The system automatically controlled summary length (50-100 words) and *plagiarism*. The summary could not be submitted when it was out of range or when it had 10 consecutive words copied from the original passage.

Each summary was treated as a document for topic modeling. The vocabulary size was 4275 after removing stop words. 6 topic models were built for different numbers of topics (4, 8, 12, 16, 20 and 24), respectively. For each model, the topic proportions and the probability sums were computed for each summary. The LDA package used for topic modeling was infer.net from Microsoft [8].

Topic proportions and probability sums were then used as document features for clustering. We used K-Mean clustering method and fixed the number of clusters to 8 for all 6 topic models.

4. Results
We define the similarity of two clustering results by

$$Sim = \frac{\sum_{i=1}^{C} number\ of\ shared\ documents\ in\ cluster\ pair\ i}{total\ number\ of\ documents}.$$

The cluster pairs were best arranged using "Hungarian Algorithm" [7] so that the similarity is the highest under the paring. For each of the two document representations, we first compared the cluster similarity between models with the number of topics 4 and 8, 8 and 12, 12 and 16, 16 and 20, and 20 and 24. We aimed to check whether or not the clusters converge as the number of topics increases.

The results showed that when the number of topics increased, clustering based on probability sum quickly converged. The similarity between 12 topics and 16 topics was 0.96. For topic-proportion-based clustering, the similarity between 8 and 12 topics went close to probability sum. However, it dropped at 12 and 16, and then went up to 0.81 for 20 and 24.

While both representations converged to some clusters, the topic-proportion-based clustering converged to the unevenly distributed clusters. The largest two clusters contained 908 documents out of 1480. In contrast, probability-sum-based clustering converged to clusters of sizes almost the same as the original summary groups.

Table 1 shows the best matched clusters to the original passages for 24-topic model. Topic-proportion-based clusters matches the original passage groups with a similarity of 0.60, whereas probability-sum-based clustering did surprisingly better. The cluster similarity to the original summary grouping was 0.98.

Table 1 Best matched clusters to original passages

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
	Topic Proportion Based Clusters							
BM	160	0	0	0	0	20	1	2
Di	6	5	101	1	0	69	0	0
EE	0	1	186	0	1	1	0	0
Fl	11	7	21	1	1	139	5	1
Hu	1	0	1	1	173	3	5	0
JM	0	0	1	0	0	179	0	1
KJ	0	0	0	0	1	1	185	1
WR	1	0	164	0	1	20	0	1
	Probability Sum Based Clusters							
BM	180	0	0	1	0	1	1	0
Di	0	176	0	0	0	6	0	0
EE	0	1	182	0	0	5	1	0
Fl	0	0	0	179	1	6	0	0
Hu	0	0	0	0	180	4	0	0
JM	0	1	0	0	0	179	1	0
KJ	0	0	0	0	1	1	186	0
WR	0	0	2	0	0	4	0	181

Note: **BM**=Butterfly and Moth, **Di**=Diabetes, **EE**=Effects of Exercising, **Fl**=Floods, **Hu**=Hurricane, **JM**=Job Market, **KJ**=Kobe and Jordan and **WR**=Working and Running.

The cluster similarity changed when the number of topics increased in topic modeling. The topic-proportion-based clustering had its highest cluster similarity 0.77 to the original grouping when the number of topics is 12. It then dropped below 0.60. The probability-sum-based clustering had higher similarities for all models than topic proportion and consistently converged toward 1.

5. ACKNOWLEDGMENTS

This research was supported by the National Science Foundation (DRK-12-0918409, 1108845), the Institute of Education Sciences (R305H050169, R305B070349, R305A080589, R305A080594, R305G020018, R305C120001), Army Research Lab (W911INF-12-2-0030), and the Office of Naval Research (N00014-00-1-0600, N00014-12-C-0643). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF, IES, or DoD.

6. REFERENCES

[1] Arora, R. and Ravindran, B. 2008. Latent Dirichelet allocation based multi-document summarization. In *Proceedings of the second workshop on Analytics for noisy unstructured text data* (Singapre, July 24 - 24, 2008). ACM, New York, NY, 91-97.

[2] Blei, D., Griffiths, T., Jordan, M., and Tenenbaum, J. 2004. Hierarchical topic models and the nested Chinese restaurant process. *Advances in Neural Information Processing Systems*, 16 (2004).

[3] Blei, D. M., Ng A. Y., and Jordan M. I. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3 (March, 2003), 993-1022.

[4] Chang, J., Gerrish, S., Wang, C., Boyd-Graber, J. L., & Blei, D. M. 2009. Reading tea leaves: How humans interpret topic models. In *Advances in neural information processing systems*. (2009). 288-296.

[5] Graesser, A. C., D'Mello, S. K., Hu, X., Cai, Z., Olney, A., and Morgan, B. 2012. AutoTutor. In P. M. McCarthy, & C. Boonthum (Eds.), *Applied natural language processing and content analysis: Identification, investigation and resolution*. Hershey, PA: IGI Global. 169-187.

[6] Kakkonen, T., Myller, N., and Sutinen, E. 2006. Applying latent Dirichlet allocation to automatic essay grading. In *Advances in Natural Language Processing*. Springer Berlin Heidelberg, 110-120.

[7] Kuhn, H. W. 1955. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*， 2 (1955), 83–97.

[8] Lau, J. H., Grieser, K., Newman, D., & Baldwin, T. 2011. Automatic labelling of topic models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* 1 (June, 2011). Association for Computational Linguistics. 1536-1545.

[9] Li, H. (2015). *The impact of pedagogical agents' conversational formality on learning and learner impressions* (Unpublished doctoral dissertation). University of Memphis, Memphis.

[10] van Heuven, W.J.B., Mandera, P., Keuleers, E., and Brysbaert, M. 2014. Subtlex-UK: A new and improved word frequency database for British English. *Quarterly Journal of Experimental Psychology*, 67 (2014), 1176-1190.

[11] Xie, P. and Xing, E. 2013. Integrating document clustering and topic modeling. In *Proceedings of the Twenty-Ninth Conference Annual Conference on Uncertainty in Artificial Intelligence* (Bellevue, Washington, USA, July 11 - 15, 2013). UAI 2013. AUAI, Corvallis, Oregon, 694-703.

Examining the necessity of problem diagrams using MOOC AB experiments

Zhongzhou Chen
Massachusetts Institute of Technology
77 Massachusetts Ave.
Cambridge, MA, 02139
617-324-2731
zchen22@mit.edu

Neset Demirci
Balıkesir Üniversitesi
Bigadiç Cd., 10145 Paşaköy
Balıkesir, Turkey
0.266.2412762
ndemirci@gmail.com

David Pritchard
Massachusetts Institute of Technology
77 Massachusetts Ave.
Cambridge, MA, 02139
617-253-6812
dpritch@mit.edu

ABSTRACT
Earlier research on problem solving suggested that including a diagram in a physics problem brings little, if any, benefit to students' problem solving success. In 6 AB experiments conducted in our MOOC, we tested the usefulness of problem diagram on 12 different physics problems, collecting over 8000 student responses in total. We found that including a problem diagram that contains no additional information very slightly improves the first attempt correct rate. On the other hand, in half of the cases, removing the diagram significantly increased the fraction of students who elected to draw their own diagram during problem solving. The results suggest that in contrast to conventional wisdom, the benefit of including a problem diagram rarely justifies the cost of creating one.

Keywords AB experiments, MOOC, problem diagrams.

1. INTRODUCTION
As instructors, we often feel obliged to accompany the problems we write with a figure or a diagram, even when all the necessary information is already included in the problem body. However in many cases, creating a "good looking" diagram or figure can be significantly time consuming and expensive. Therefore, it is a valuable question to ask whether a problem diagram does indeed help students solve problems more accurately or more quickly, and if so, does the benefit justify the cost of creating one?

Cognitive learning theories, such as dual coding hypothesis [7] and multimedia learning theories [6, 8] indirectly suggest, that diagrams can be potentially beneficial to problem solving. On the other hand, a series of recent experiments by Lin, Maris and Sigh[2–4]found that for the problems involved in their study, the accompanying diagrams have no detectable benefit for problem solving, and sometimes hurt performance by discouraging students to draw their own diagrams during problem solving.

Using the "split test" feature of the edX platform [1], this study addresses the following research questions in the context of a calculus based introductory mechanics course:

1. Do diagrams in general have an impact on students' problem solving performance (either percentage of correct answer or time spent on problem solving)? If so, to what extent?

2. Do diagrams change students' problem solving behavior, or more specifically, their decision to draw their own diagram?

2. MATERIALS AND METHODS

2.1 AB experiment on the edX platform
The edX platform allows the course creator to create controlled AB experiments by splitting the student population into two or more groups (called "partitions"), and presenting each group with a different version of content, such as a problem or a series of problems and html pages. Every student who tries to access the experimental course content for the first time is randomly assigned to one of the groups at the time of the access.

2.2 Experiment Design
A total of six experiments with identical design were implemented throughout the first eight units of the course. Each experiment involves two problems chosen from either the homework or the quiz section of a given unit, so the study involves twelve different problems in total. The problems were chosen from the first eight units of the course, covering kinematics, Newton's laws, circular motion, conservation of momentum, and conservation of energy.

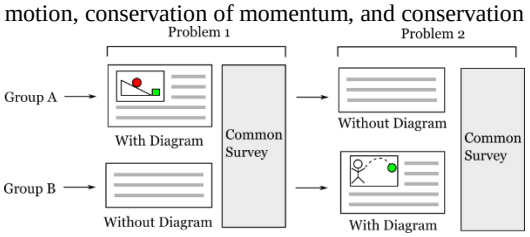


Figure 1: Experiment design. Each experiment consists of a pair of problems differing only in whether (DG) or not (NDG) they had a diagram. The same design is used for all 6 experiments conducted.

In each two-problem experiment, the student population was randomly partitioned into two groups: A and B (**Figure 1**). Group A saw the first problem in DG format and the second problem in NDG format. Group B saw the two problems in the same order, but the DG/NDG condition was reversed. The group assignment for each experiment is independent, reducing systematic bias.

Depending on when each experiment was released to students in the course, the number of students in each group ranged from ~480 (week 2) to ~180 (week 7).

The following survey question was asked after each problem:

When solving this problem, (check all that apply)

- ○ I drew one or more diagrams
- ○ I wrote down some equations
- ○ I did the problem entirely in my head
- ○ I used some other means to solve the problem

Only students who answered both the problem and the survey were included in the analysis.

3. RESULTS AND DISCUSSION

3.1 Results

We first look at the impact of including a diagram on the percentage of correct answer on students' first attempt. In most cases (see Fig 2 below) the presence or absence of a diagram has little impact on the difficulty of the problem itself. Only 3 out of 12 problems (P3, P4 and P8) showed a significant difference in difficulty between the two conditions ($p < 0.05, \chi^2 > 5$).

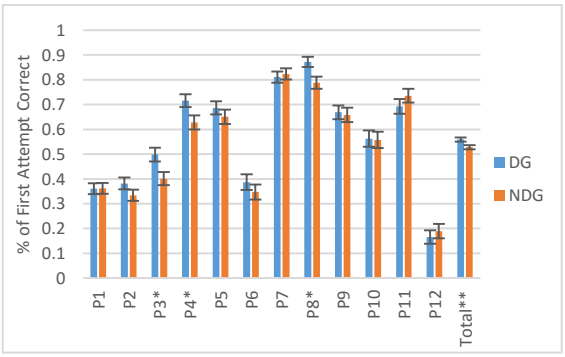


Figure 2: Percentage of first attempt correct for each problem. *Difference is significant at the 0.05 level. ** Difference is significant at the 0.01 level. (Chi-squared test)

Since we carefully balanced systematic bias in the population in our experiment design, it is meaningful to add up the data from all 12 problems and compare the overall success rate between the DG vs. NDG conditions (rightmost column in Fig 2). The overall correct rate under the DG condition is higher than that in the NDG condition by 3 ± 0.8 %. The difference, although small, is still statistically significant due to the large cumulative sample size (~ 3500 observations per condition, $p < 0.01, \chi^2 = 6.9$).

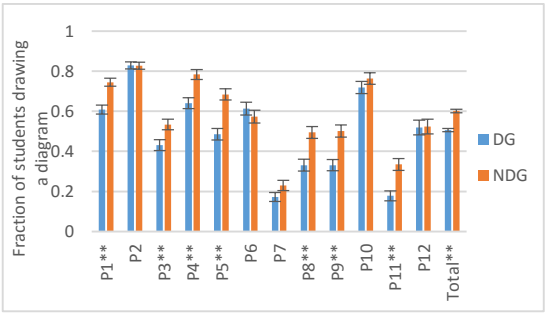


Figure 3: Percentage of students who drew a diagram solving each problem. *Difference is significant at the 0.05 level. ** Difference is significant at the 0.01 level. (Chi-squared test)

The presence/absence of a problem diagram impacts students' tendency to draw their own diagram as measured by the survey question. As shown in Figure 3, on 7 out of 12 problems, a significantly lower fraction of students ($p < 0.01, \chi^2 > 7$, Chi-square test) in the DG condition reported drawing their own diagram during problem solving than in the NDG condition. A noteworthy observation (Fig. 3) is the high variation in sensitivity of different problems to the DG/NDG condition. Combining the data across all 12 problems, students in the DG condition are 10% less likely to draw their own diagram than in the NDG condition ($p < 0.001, \chi^2 = 65$).

3.2 Discussion

Perhaps the most surprising observation of this study is how little students benefit from a problem diagram. Even with the large sample size provided by MOOC, significant difference between the two conditions are only observed for 3 out of 12 problems, with the largest difference at 10% and the overall difference at merely 3%.

Those results suggest that even though the benefits predicted by conventional wisdom and dual-coding hypothesis may still exist, the effect size might be small in an *in vivo* situation and only significant in the more extreme cases. For the majority of "normal" physics problems, our findings are consistent with previous studies [2–5] indicating that the benefit of a diagram is small.

In stark contrast to the correct rate, the decision to draw is very sensitive to the DG/NDG condition on 7 out of 12 problems: when the problem diagram is removed, students are 10% more likely to draw their own.

For instructors, the study suggests that for common physics problems of average difficulty, the benefit of adding a diagram may be too small to justify the resource and effort required to create it.

4. ACKNOWLEDGMENTS

Our thanks to Dr. Qian Zhou for helping on data analysis.

5. REFERENCES

[1] edX Documentation: Creating Content Experiments: *http://edx.readthedocs.org/projects/edx-partner-course-staff/en/latest/content_experiments/index.html*.

[2] Lin, S.-Y. et al. 2013. Student difficulties in translating between mathematical and graphical representations in introductory physics. 250, (2013), 250–253.

[3] Maries, A. and Singh, C. 2014. A good diagram is valuable despite the choice of a mathematical approach to problem solving. *2013 Physics Education Research Conference Proceedings*. (Feb. 2014), 31–34.

[4] Maries, A. and Singh, C. 2012. Should students be provided diagrams or asked to draw them while solving introductory physics problems? *AIP Conference Proceedings*. 1413, (2012), 263–266.

[5] Maries, A. and Singh, C. 2013. To use or not to use diagrams: The effect of drawing a diagram in solving introductory physics problems. *AIP Conference Proceedings*. 1513, 1 (2013), 282–285.

[6] Mayer, R.E. 2001. *Multimedia Learning*. Cambridge University Press.

[7] Paivio, A. 1986. *Mental representations: a dual coding approach*. Oxford University Press.

[8] Schnotz, W. 2002. Towards an Integrated View of Learning From Text and Visual Displays. *Educational Psychology*. 14, 1 (2002), 101–120.

Identifying relevant user behavior, predicting learning, and persistence in an ITS-based afterschool program

Scotty D. Craig
Arizona State University
scotty.craig@asu.edu

Xudong Huang
The University of Memphis
xhuang3@memphis.edu

Jun Xie
The University of Memphis
jxie2@memphis.edu

Ying Fang
The University of Memphis
yfang2@memphis.edu

Xiangen Hu
The University of Memphis
xhu@memphis.edu

ABSTRACT

ALEKS (Assessment and Learning in Knowledge Spaces) has recently shown promise for effectively training mathematics at equivalent levels to human teachers. However, not much is known about how the system accomplished this. In this paper, we describe the use of three data mining techniques used to analyze student data from an afterschool program with ALEKS. Our first analysis used DMM modeling and k-clustering to identify important groups of behaviors within ALEKS users and to show the importance of context for elements. Our second analysis focused on identifying learner behaviors that predict student learning during the program. The final analysis presents a method for determine learner persistence within the afterschool program.

Keywords

ALEKS, Afterschool programs, learning strategies, help seeking, persistence

1. INTRODUCTION

ALEKS is a web-based learning system with artificial intelligence components that are based in Knowledge Space Theory [1]. Instead of giving scores to measure a student's overall mastery of the subject, the theory allows for a precise assessment of what the student knows, does not know, and is ready to learn next. The probability of mastery for a knowledge state increases as students correctly answer questions containing that problem type.

ALEKS is a highly effective educational technology program shown to perform at the same level as other major ITS systems in mathematics [2]. In a four year evaluation of ALEKS in an afterschool setting, the students tutored by ALEKS or taught by expert teachers in one after-school program showed the same level of performance in a mathematics state test [3,4], and outperformed controls not participating in the program[5].

1.1 Current investigation

1.1.1 ALEKS afterschool program

The afterschool program was implemented for 25-week after school. It was held twice a week for 2 hours each day. Students received three 20-minute learning segments with a 20-minute break between each. Student logs were recorded by ALEKS. The students were from five middle schools in west Tennessee. The schools were located in a mid-sized city and the surrounding rural area, having a largely economically disadvantaged population (68.2%) and large minority student enrollment (56.3% African American, 39.3% White, and 4.4% others). None of the five schools reach an average SES level of Tennessee (i.e., 54.4% of the students eligible for free or reduced-price lunch).

1.1.2 Research question

While the afterschool program demonstrated that students using ALEKS could perform at the same levels as student in teacher-led classrooms [3.5], the student's learning process that led to this result is still unclear. Summaries of three methods are presented to show how popular data mining techniques can be applied to ALEKS log files to better understand student's behavior in the ALEKS afterschool program.

2. Learning strategies with DMM

There are distinct advantages for analyzing sequences over raw frequencies. The frequency counts could indicate that the two students used the same strategy. However in context, the two students act differently because the patters have different sequences. Modeling learning sequences is not as direct as frequency counting. One way to measure sequence is to calculate similarities in sequences, and then cluster the sequences using the similarities. A method, modeling learning sequences with Discrete Markov Models (DMM) and clustering with a k-means algorithm, has successfully discovered help-seeking strategies in ITS [6].

The analysis used 55,281 learning sequences of 372 students on ALEKS system. Typical activities students made include: correct, wrong, explain, mastery (added to pie), failed, and left the attempt. We recoded the same actions in a row as action - action2 - action3 – action3 for easy interpretation.

With DMM modeling and k-means clustering for all transitions, ten learning strategies emerged. These strategies were Cluster 1 – three correct practices in a row and reach mastery (9%), Cluster 2 – Quick mastery (11%), Cluster 3 – keep practice after mastery (6%), Cluster 4 – Frequently request worked examples and only try when confident (7%), Cluster 5 – Request worked examples after wrong and get correct and mastery finally (12%), Cluster 6 – Request worked examples then quit without practice (13%), Cluster 7 – Request worked examples after wrong but still get wrong then quit (17%), Cluster 8 – Correct at 1st practice but wrong at 2nd & 3rd, then request worked examples but only get half practices correct then. (6%), Cluster 9 – All practice are wrong, request worked example after 2 wrongs, still get wrong, quit or reach failure. (9%), and Cluster 10 – All practice are wrong, reach failure and then 2nd failure (9%).

3. Learning behaviors and learning outcome

A sample from 204 students was used to predict students learning using behaviors within ALEKS. The learning behaviors recorded in ALEKS log files were categorized into help-seeking and practice. We utilized logistic mixed effects models to investigate the relationship of help-seeking and practice with learning outcome. Topics and students were random variables. The model also included student's pretest which was measured by 5th grade TCAP score. The learning outcome was topic mastery (1 or 0).

3.1 Help-seeking and learning outcome

The results of logistic mixed effects model indicated four significant help-seeking behaviors were predictive of learning (R^2 = .81, For full results See Table 1). We used 10-fold cross validation to validate the mixed effects model of help-seeking.

Table 1.
Student help-seeking behaviors that predict learning outcomes

Learning behaviors	Coefficient	Std. Err	z	p
Pretest	.35	.08	4.32	.000
Reading Explain first	.42	.14	3.12	.00
Proportion explain	-46.86	1.51	-31.13	.000
Explain after mistake	-.36	.35	-1.05	.29
Explain request latency	-.01	1.29	27.79	.000
Explain avoid mistake	35.99	.01	-2.40	.02

3.2 Practice and learning outcome

The results of logistic mixed effect model indicated five significant patters of making mistakes were related to learning (R^2 = .75, See Table 2 for results). A 10-fold cross validation was adopted to validate the mixed effects model of practice.

Table 2
Student practice behaviors that predict learning outcomes

Learning behaviors	Coefficient	Std. Err	z	p
Pretest	.17	.10	1.64	.10
Initial Mistake	.64	.09	7.23	.000
Mistake (%)	-5.35	.32	-16.85	.000
Success (%)	12.65	.49	26.04	.000
Self-correction	-1.3	.24	-5.52	.000
Self-correction time	.01	.003	2.23	.03

4. Prior knowledge, difficulty on persistence

A sample from 114 student log files utilizing 92,235 lines of log files data from years two and three of the program that included date, time, topics attempted and the result of each trial were used to predict student's persistence using prior knowledge topic difficulty and time period. The number of trials (T) was chosen as the measure of persistence. Then, three levels of persistence were defined: high persistence (T>15), medium persistence (10<=T<15), and non-persistence (T<5 and not reach mastery).

4.1 Results

Logistic regressions were performed to explore the effects of prior knowledge, topic difficulty and time period the learning took place on the likelihood of participant's persistence related behavior. For high persistence, the model was significant, $\chi 2(3) = 124.14$, $p < .001$, explaining 2.8% (Nagelkerke R^2) of the variance of highly persistence students and correctly classified 96.2% of cases. For medium persistence, the model was significant, $\chi 2(3) = 118.68$, $p < .001$, explaining 1.8% (Nagelkerke R^2) of the variance in medium persistence and correctly classified 93.3% of cases. Increasing topic difficulty was associated with increased persistence, but increasing prior knowledge and days learning in the system was associated with a reduction in persistence. For non-persistence, the model was statistically significant, $\chi^2(3) = 864.88$, $p < .001$, explaining 6.8% (Nagelkerke R^2) of the variance in non-persistence and correctly classified 62.5% of cases. Increasing topic difficulty was associated with an increased non-persistence. Increasing prior knowledge was associated with a reduction in non-persistence.

5. Discussion/conclusion

The current paper present three methods to analyze learner performance which identify important clusters of learner strategies during learning with ALEKS, help seeking behaviors that predict learning, and persistence. The first analysis clustered learner strategies and demonstrated that context is important when looking at clusters. Thus identical elements or techniques can serve different functions when the sequence occurs at a different point in the learning process. The second two analyses use features from the ALEKS data logs to predict learning and persistence. The second analysis found that latency to seek help was negatively related to mastering a topic. This is a validation that ALEKS is working in that increase practice with the system was predictive of mastery of topics. For student persistence, while predicted variability was small, the models were very reliable and able to classify a large proportion of the data. The pattern of data for non-persistent behavior was interesting finding that lower prior knowledge students work on problems projected to be of greater individual difficulty which is predictive of lower persistence. Taken together these techniques indicate patterns that are easily detected and corrected within systems like ALEKS.

6. ACKNOWLEDGMENTS

This research was supported by the Institute for Education Sciences (IES) Grant R305A090528.

7. REFERENCES

[1] Falmagne, J.C., Koppen, M., Villano, M., Doignon, J.P. and Johannesen, L., 1990. Introduction to knowledge spaces: How to build, test, and search them. *Psychological Review, 97*(2), 201.-224.

[2] Sabo, K.E., Atkinson, R.K., Barrus, A.L., Joseph, S.S. and Perez, R.S., 2013. Searching for the two sigma advantage: Evaluating algebra intelligent tutors. *Computers in Human Behavior*, 29(4), 1833-1840.

[3] Craig, S. D., Hu, X., Graesser, A. C., Bargagliotti A. E., Sterbinsky, A., Cheney, K. R., & Okwumabua, T. 2013. The impact of a technology-based mathematics after-school program using ALEKS on student's knowledge and behaviors. *Computers & Education*, 68, 495-504.

[4] Huang, X., Craig, S.D., Xie, J., Graesser, A. and Hu, X., 2016. Intelligent tutoring systems work as a math gap reducer in 6th grade after-school program. *Learning and Individual Differences*, 47, 258-265.

[5] Hu, X., Craig, S.D., Bargagliotti, A.E., Graesser, A.C., Okwumabua, T., Anderson, C., Cheney, K.R. and Sterbinsky, A., 2012. The Effects of a Traditional and Technology-based After-school Setting on 6th Grade Student's Mathematics Skills. *Journal of Computers in Mathematics and Science Teaching*, 31(1), 17-38.

[6] Vaessen, B.E., Prins, F.J. and Jeuring, J., 2014. University students achievement goals and help-seeking strategies in an ITS. *Computers & Education*, 72, 196-208.

Extracting Measures of Active Learning and Student Self-Regulated Learning Strategies from MOOC Data

Nicholas Diana
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
ndiana@cmu.edu

Michael Eagle
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
meagle@cs.cmu.edu

John Stamper
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
john@stamper.org

Kenneth R. Koedinger
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA 15213
koedinger@cmu.edu

ABSTRACT
Previous work has demonstrated that in the context of Massively Open Online Courses (MOOCs), doing activities is more predictive of learning than reading text or watching videos (Koedinger et al., 2015). This paper breaks down the general behaviors of reading and watching into finer behaviors, and considers how these finer behaviors may provide evidence for active learning as well. By characterizing learner strategies through patterns in their data, we can evaluate which strategies (or measures of them) are predictive of learning outcomes. We investigated strategies such as page re-reading (active reading) and video watching in response to an incorrect attempt (active watching) and found that they add predictive power beyond mere counts of the amount of doing, reading, and watching.

Keywords
MOOCs, active learning, self-regulated learning

1. INTRODUCTION
The growing popularity of MOOCs has prompted an examination of the effectiveness of prototypical MOOC activities such as watching video lectures. Most recently, Koedinger et al. (2015) explored the impact of watching video lectures, reading course content, and doing interactive activities. They found that doing activities had a larger impact than reading course content or watching videos. The authors attribute this effect, at least in part, to the fact that doing activities is necessarily an active form of learning, whereas reading content and watching videos is generally passive.

However, not all *reading* and *watching* is done passively. This study returns to the dataset used in Koedinger et al. (2015) and attempts to extract new features that are representative of different types of active learning behaviors and student strategies. By exploring these finer-grained measures of student behavior, we are able to: 1) support the results of Koedinger et al. (2015) by providing more evidence that active learning behaviors are associated with better learning outcomes, and 2) demonstrate that evidence of active learning can not only be mined from *doing* data, but from *reading* and *watching* data as well.

2. BACKGROUND
2.1 Previously Explored Features
Koedinger et al. (2015) designed three features to capture *doing, watching*, and *reading* behavior within a MOOC. *Doing* behavior was characterized by the total number of activities started throughout the course. *Watching* behavior was characterized by the number of times the user clicked play while viewing a video in the MOOC (referred to by the feature name "video"). In this count, consecutive plays of the same video were not counted.

The course content and interactive activities often appeared on the same page, so estimating a measure of *reading* behavior was slightly more complex. *Reading* was estimated using a ratio of about 3.4 activities per page, and then subtracting pages viewed for activity access from total pages viewed. While not as precise as some other measures, the goal of this measure is to capture variation in student reading.

Left unexplored are more complex features dependent on patterns of actions. We build off of the features previously explored in Koedinger et al. (2015) to generate features representative of student strategies embedded in watching and reading data.

2.2 Finer-grained Features
With respect to *watching* behavior, we extended beyond raw counts and instead looked at possible interactions between *watching* and *doing*. We hypothesized that students who complete problems while watching videos, and students who reference videos after incorrect attempts do better on the final exam. For *reading* behavior, we examined the impact of the common, albeit surface-level strategy of reviewing a page to re-read content [1,2], hypothesizing that students who review content do better on the final exam.

3. DATA AND METHOD
3.1 Data
The data used are from a 12-week survey course titled "Introduction to Psychology as a Science." The lectures, along with slides, a discussion form, quizzes, and exams, were provided via Coursera. The Open Learning Initiative (OLI) Learning Environment was embedded into Coursera to provide readings and interactive activities.

The current study used a subset of this dataset, which contains only students who registered for the OLI portion of the course and took the final exam (N=939). On average each student generated 2757 transactions, though the actual number varied greatly among students (SD=1909). This dataset is freely available (with administrator permission) via the online learning data repository and analysis service, DataShop [3] at:

https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=863.

3.2 Model Building
To understand the impact of the new features on learning outcomes relative to the previously explored features, a linear regression model was generated that included the three original *watching, reading,* and *doing* features. This model serves as a baseline. A new linear model was generated for each new feature. The new feature was added alongside the previously explored features to predict final exam score, unless it was redundant with another feature.

4. RESULTS AND DISCUSSION

4.1 Baseline Model

As expected, the baseline model showed that the *doing* measure had a high impact on final exam performance ($p<.001$). Neither the *reading* nor the *watching* measures were significant predictors. The results of this model can be seen in Table 1 in the row labeled "Baseline."

4.2 Watching

4.2.1 Attempting Activities During Video Playback

We hypothesized that some students may be watching videos and doing activities simultaneously, potentially answering questions as the relevant material is covered in the video lecture. To test this hypothesis, we extracted a new feature that represents the proportion of all activity attempts that occurred during video playback. When added to the baseline model, the proportion of attempts that occurred during video playback was predictive of final exam performance, though marginally significant ($p<.1$). This may indicate that some students are answering problems while watching relevant videos, and that this is a successful strategy. The results of this model can be seen in Table 1 in the row labeled "% attempts during playback."

4.2.2 Referencing Videos After Incorrect Attempts

We similarly hypothesized that some students may reference the video lectures after an incorrect attempt on an activity. To test this, we extracted a new feature representing the proportion of all video play actions that occurred after an incorrect attempt, but before the next attempt on the same problem. When added to the baseline model, the proportion of video play actions that occurred between attempts on the same problem was predictive of final exam performance, though again, marginally significant ($p<.1$). This may indicate that some students are referring back to videos to find correct answers. The results of this model can be seen in Table 1 in the row labeled "% plays after incorrect attempts."

4.3 Reading

4.3.1 Only-Reading Page Views

In the current version of OLI course content and activities appear on the same page. To compensate for this, we counted the number of pages viewed without any activity attempts. To mitigate pages viewed quickly on the way to another page, we eliminated any page viewed less than 10 seconds from this count. When added to the baseline model (with "non-activity page views" removed for redundancy), the number of only-reading page views is predictive of final exam performance ($p<.05$). The results of this model can be seen in Table 1 in the row labeled "Only-reading page views." Note that this is by no means a complete measure of all *reading* behavior because it misses any reading done on pages where the student also attempted activities.

4.3.2 Re-reading Page Views

We also found that, when added to the baseline model (again with "non-activity page views" removed for redundancy), the number of second page views that are reading only page views (i.e., pages revisited with 0 activity attempts) is predictive ($p<.001$). This suggests at least some students review material by re-reading course content, and that this strategic reading is predictive of final exam performance. The results of this model can be seen in Table 1 in the row labeled "pages re-read."

5. CONCLUSION

Our work examines how evidence of active learning can be extracted from *reading* and *watching* data as well as *doing* data, and demonstrates that these measures can be predictive of learning outcomes. Re-reading pages (a measure of active reading) and attempting activities while watching videos (active watching) improved prediction of learning outcomes beyond the simple measure of active doing. While more research is needed to test their generality, these features may help establish a more nuanced characterization of learner strategies.

6. REFERENCES

[1] Alexander, P. (1997). Mapping the multidimensional nature of domain learning: The interplay of cognitive, motivational, and strategic forces. In M. L. Maehr & P. R. Pintrich (Eds.), *Advances in motivation and achievement*, 213–250.

[2] Azevedo, R., Johnson, A., Chauncey, A., Graesser, A., Zimmerman, B., & Schunk, D. (2011). Use of hypermedia to assess and convey self-regulated learning. *Handbook of self-regulation of learning and performance*, 102-121.

[3] Koedinger, K.R., Baker, R.S.J.d., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A Data Repository for the EDM commuity: The PSLC DataShop. In Romero, Ventura, Pechenizkiy, Baker, (Eds.) *Handbook of Educational Data Mining*. CRC Press.

[4] Koedinger, K. R., Mclaughlin, E. a, Kim, J., Zhuxin Jia, J., & Bier, N. L. (2015). Learning is Not a Spectator Sport: Doing is Better than Watching for Learning from a MOOC, L@S (Mar. 2015), 111–120. DOI=http://doi.org/10.1145/2724660.2724681

Table 1. Linear regression models that include new features.

Added Feature	Activities Started	Non-Activity Page Views	Video	Added Feature(s)	RMSE	Adj. r^2	AIC
N/A (baseline)	1.8206***	0.3632	0.1509	-	6.768	0.0785	6261.855
% attempts during playback	1.8990***	0.2776	0.2241	0.3753.	6.472	0.0781	5541.207
% plays after incorrect attempts	1.9263***	0.2653	0.1361	0.3845.	6.66	0.0811	5986.356
Only-reading page views	1.7775***	-	0.1458	0.5129*	6.759	0.0808	6259.458
Pages re-read	1.5436***	-	0.1437	0.8468***	6.736	0.0871	6253.016

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 .' 0.1 ' ' 1

Exploring Social Influence on the Usage of Resources in an Online Learning Community

Ogheneovo Dibie
Institute of Cognitive Science
Dept. of Computer Science
University of Colorado
Boulder, Colorado USA
ogdi0204@colorado.edu

Tamara Sumner
Institute of Cognitive Science
Dept. of Computer Science
University of Colorado
Boulder, Colorado USA
sumner@colorado.edu

Keith Maull
University Corporation for
Atmospheric Research
Boulder, Colorado USA
kmaull@ucar.edu

David Quigley
Institute of Cognitive Science
Dept. of Computer Science
University of Colorado
Boulder, Colorado USA
david.quigley@colorado.edu

ABSTRACT

This research investigates the usage distribution of instructional resources shared among educators in an online learning community. The usage of a resource is defined by the number of unique educators who use (click on) it. We explored what the usage distribution of these resources looks like and we investigated what underlying mechanisms may have generated the observed distribution. Our results indicate that the usage distribution of resources follows a power law. Furthermore, our results also suggest that an educator's decision to use a resource may be influenced by the prior decisions of others. 82.6% of 2500 simulations of an information cascade model developed to model the resource selection process of educators resulted in a power law distribution as observed in our data. Information cascades provide a natural way of understanding how individuals may imitate the decisions of others even when such decisions do not align with their perssonal preferences.

1. INTRODUCTION

Research consistently indicates that online learning communities can improve the instructional practices of educators and produce increases in student learning outcomes by providing educators with access to learning resources and best practices shared by their peers [5]. Given the importance of these community-contributed resources to educator instruction, understanding the factors that encourage their usage is an intriguing question with important implications for educator instruction, student learning and agencies that support these communities.

We explored this question in the context of a community of Earth Science educators that used an online curriculum planning tool called the Curriculum Customization Service (CCS). The CCS provides educators with access to digital versions of their class textbook, digital library resources and community-contributed resources. This study is based on 6th-9th grade Earth Science educators that shared and used community-contributed resources in the CCS over a period of four academic years.

We began by exploring the observed usage distribution of community-contributed resources in the CCS, and then turned our attention to the influence of three mechanisms on the observed usage distribution. First, we investigated how resource visibility influences resource selection—postulating that the position or rank of a resource in the list it is displayed in *may* impact selection behavior. Second, we investigated how the quality (or perceived quality) of a resource might have influence on selection. Finally, we examine how social factors, specifically how the decisions of others in the community, provide insights into the observed resource usage distribution.

2. METHODS AND RESULTS

We discovered that the usage distribution of community-contributed resources follows a power law. Also known as Zipf, Pareto-Levy or scale-free distributions [4], a quantity x obeys a power law if it is drawn from a probability distribution $p(x) \propto x^{-\alpha}$ where α is known as the exponent or scaling parameter. Power laws appear in a wide array of man made and natural phenomena [3] such as the distribution of calls to telephone numbers, scientific paper citations and the frequency of use of English words [4]. We determined that the usage distribution of resources followed a power law using software implementations[1] [2] of the rigorous statistical approach of Clauset et al. [3] for detecting power laws in empirical data. [3]. Our empirical data was found to follow a power law with an α of 4.44 and an x_{min} value of 15. Figure 1 illustrates that a power law provides a closer fit to the complimentary cumulative distribution function (CCDF) [3]

[1] plfit: https://pypi.python.org/pypi/plfit
[2] powerlaw: https://pypi.python.org/pypi/powerlaw
[3] The CCDF is defined by $\Pr(X \geq x)$

of the empirical data in comparison to the lognormal and exponential distribution.

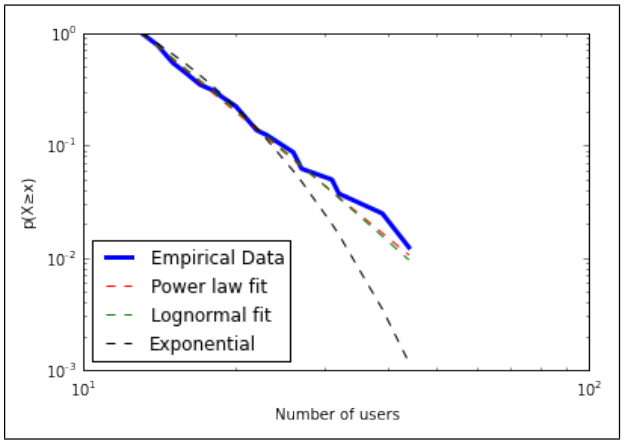


Figure 1: Comparisons of the complimentary cummulative distribution function (CCDF) of the empirical data, the power law, lognormal and exponential distribution fits to the data.

2.1 Mechanisms behind the power law distribution of resources

Resource position: Correlation tests between the mode, median and last click position of resources and their usage show only a very weak correlation between the usage of a resource and its position during the 2012-2013 school year. This suggests that a resource position had little to no influence on usage.

Resource quality: We then investigated the relationship between resource quality (inferred before a user clicks on it) and its usage in two steps. First, we used the presence of a description in the listing of a resource as a marker of its quality. Thus, resources with a description were deemed as having high quality and those without a description were regarded as low quality. We then investigated if there was a statistically significant difference in usage between resources of high quality and those of low quality, and consequently discovered no statistically significant difference in usage between resources of both groups. Our next investigation into the impact of a resource's quality on usage investigated whether there was any correlation between the number of quality signals of a resource and its usage. To do this, we developed a composite resource quality score that incorporated all signals of a resource's quality that can be inferred by a user before clicking. These signals were mapped to the resource quality indicators developed by Bethard et al. [1]. Our results show only a weak correlation of 0.124 between resource quality and usage ($t = 2.8343, df = 516, p = 0.002387$)

Social influence: Finally, we looked at the impact of aggregate social influence on the usage of community-contributed resources. We found a statistically significant positive correlation of 0.634 at a p-value of $2.2e^{-16}$ between saves and usage. Unlike our earlier tests on position and quality, this indicates that the social influence conveyed through the saving of resources may be in part responsible for driving usage.

We then explored if an information cascade model simulating the decision making processes of educators can generate a power law usage distribution as observed in our data. Our model extends the informational cascade model of Bikchandani, Hirshleifer, and Welch (BHW) [2] in three ways. First, instead of the binary decision model of BHW, a decision will be made between $1..r$ resources at any time. Second, in contrast to the BHW model, the decision of an individual is not always visible to others as a public signal. In our context, the only public signal available is whether or not a user saves a resource. After clicking on a resource, users will leave public signals with a uniform random probably p. This probability is exogenously fixed at 0.41—determined from computing the ratio of saves to unique clicks on all resources across all school years. Finally, a user's private signal p_s for a resource r is drawn from a discrete uniform probability distribution such that $p_s \in [0, 1]$. 2500 simulations of the information cascade model described above were processed with each simulation evaluated to see if they follow a power law using the procedure of Clauset et al. [3]. Consequently, 82.6% of these simulations were determined to follow a power law distribution. The outcomes of this experiment strongly suggests that an information cascade model simulating the decision making process of educators can lead to a power law usage distribution as observed in our data. This provides strong support for the social influence hypothesis as a generative mechanism for the observed usage distribution.

3. DISCUSSION & CONCLUSION

For agencies that support online learning communities, this research has important implications for resource presentation and recommendation. In presenting resources, social influence signals can be de-emphasized to limit the chances that they will detract users from evaluating a resource's inherent quality. For example, in the CCS, the number of educators that have saved a resource can be hidden and require active effort from users to be revealed. In recommending resources, high quality but barely used resources can be recommended to educators in ways that give them high precedence. This could include personalized recommendations while active on the platform or email recommendations. This paper is based upon research supported by National Science Foundation awards #1043638 and #1147590.

4. REFERENCES

[1] Bethard, S., Wetzer, P., Butcher, K., Martin, J. H., and Sumner, T. Automatically characterizing resource quality for educational digital libraries. In *Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries* (2009), ACM, pp. 221–230.

[2] Bikhchandani, S., Hirshleifer, D., and Welch, I. A theory of fads, fashion, custom, and cultural change as informational cascades. *Journal of political Economy* (1992), 992–1026.

[3] Clauset, A., Shalizi, C. R., and Newman, M. E. Power-law distributions in empirical data. *SIAM review 51*, 4 (2009), 661–703.

[4] Easley, D., and Kleinberg, J. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2010.

[5] Louis, K. S., and Marks, H. M. Does professional community affect the classroom? teachers' work and student experiences in restructuring schools. *American journal of education* (1998), 532–575.

Time Series Cross Section method for monitoring students' page views of course materials and improving classroom teaching

Konomu DOBASHI
Faculty of Modern Chinese Studies
Aichi University
4-60-6 Hiraike-cho Nakamura-ku
Nagoya-shi Aichi-ken
453-8777 Japan
dobashi@vega.aichi-u.ac.jp

ABSTRACT
To enable teachers to monitor student engagement and improve classroom instruction, a data mining method and an Excel macro are developed in this work. The data mining method is based on a Time Series Cross Section (TSCS) framework and designed for application to students' page views of course materials that are created over Moodle. The Excel macro generates TSCS tables of students' page views and reflect the viewing behaviors of students over time as transitioning numerical values.

Keywords
Time Series, Cross Section, page views student engagement, educational data mining

1. INTRODUCTION
A teacher is responsible for ensuring proper delivery of lessons in the classroom while simultaneously understanding the individual reactions and progress of students. Effectively satisfying these roles are essential to improving the quality of education. The problem is that in a class comprising dozens of students, accurately measuring individual reactions and progress is difficult even for experienced teachers. Another challenge is how such data can be provided to both educators and learners. A favorable strategy is to supply teachers with the results of appropriately conducted analyses in a timely manner so that analytical insights can be used to advance teaching enhancement. A tool that can be employed frequently in class for such purpose is equally desirable. We propose an Excel macro that semi-automatically generates TCSC tables from Moodle logs. The system monitors and records the time that students spend on browsing and their page views in class. It also provides data and suggestions that can be used as reference for reinforcing classroom instruction and keeping track of student engagement.

2. RELATED RESEARCH
Currently, analyzing Moodle logs [6] is primarily based on Excel or CSV data. Because the macro developed in this study is grounded in Excel and pivot table functions, teachers can easily obtain the summaries of the frequency at which students view course materials [1, 2]. Moodog [7] that Zhang and Almeroth has developed that incorporated an analysis function of log in Moodle. This system is able to analyze the course materials browsing rate, page views and viewing time of students. The analytical results are displayed on the Moodle screens, it represents interaction of the students and Moodle using graphs and the tables.

Mazza and Dimitrova has been developed a system called CourseVis [3] that to track the student's behavior in an online class, it can be visualized by the graph along the access status to the content page to the course schedule. Also Gismo [4] also take advantage of the access history of Moodle and visualized using the access graph to the students of the courses and teaching materials, it is to understand the behavior of the students. In the current it has been provided so that it can be installed as part of the Moodle.

Google Analytics [2] provides a website analysis service that enables data analyses grounded in different perspectives. Such service also helps educators improve course materials and lessons. Whereas Google Analytics can be used only by a Moodle administrator, the method proposed in this paper can be employed by any Moodle user. The developed macro is equally accessible to any Moodle course administrator.

3. METHOD AND EXPERIMENTS
Excel has several features designed to process qualitative data. Among these, the pivot table feature enables users to count qualitative data, such as strings; create a cross section table; and quantify input data. These functions were applied in this work. The tabulation generated in this study is referred to as a "Time Series Cross Section table" because an aggregated pivot table was created to incorporate time series data into the analysis.

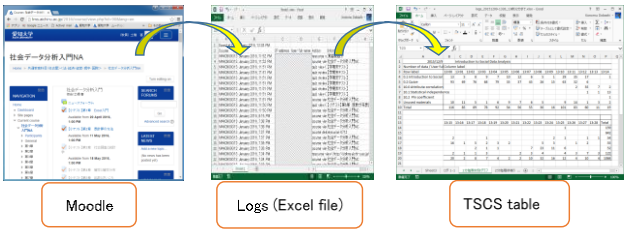


Figure 1. Overview of processing

This study primarily used PDF files that are viewable through a PDF viewer by clicking on a link in the table of contents created in the Moodle topics format, which is commonly used in Moodle based courses. The TSCS tables generated in this research features columns on time, user full name, action, and information. These

data are aggregated by using Excel's statistical functions and pivot table features to semi-automatically generate the TSCS tables.

While delivering a lesson, the teacher can assess student status and if necessary, download Moodle logs to a specified folder and run the macro. Downloading of logs and macro processing take only tens of seconds. These features guarantee that sufficient time and focus is devoted to a lesson. After a lesson is completed, the teacher can run the macro (if necessary) without having to worry about processing time during a class.

The developed macro was applied in the Introduction to "Social Data Analysis" class offered at the case university to demonstrate how a TSCS table is generated. Table 1 is the TSCS table of page views for course items (1-minute intervals). On December 9, 2015, the teacher discussed the lesson on attribute correlation for 90 minutes. The lesson was initiated at 13:00 and ended at 14:30. Table 1 shows the TSCS table generated at 1-minute intervals, downloaded at 13:28 from Moodle logs, and aggregated. Page views of the course items were counted from the beginning of the lesson up to 13:28 (Table 1).

The TSCS table for students (generated 2-minutes intervals) shows that viewing was concentrated from 13:12 to 13:16 and at 13:24 (Table 2). Some students exhibited a delay in accessing the materials at 13:18, 13:20, 13:26, and 13:28. With a TSCS table for each student, the teacher can determine which students are viewing materials and which have recently browsed the materials (Table 2).

Table 1. Example of TSCS table for course items generated 2-minutes intervals

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Number of data / U	Column label															
2	Row label	13:02	13:04	13:06	13:08	13:10	13:12	13:14	13:16	13:18	13:20	13:22	13:24	13:26	13:28	13:30	Total
3	0.1 Introduction to Social Data	18	17	17	18	6	64	17	1					1			159
4	0.2 Quize	162	144	114	80	41	95	6									642
5	10.0 Attribute correlation							2	42	4		1		3	1	1	54
6	10.1 Statistical Independence								2	29	6	5	2	5	3	3	55
7	10.2 Phi coefficient									1	2	2		30	17		52
8	Unused materials	21	6	15	13	5	23	4	4	2	3	2	7	4	10	3	122
9	Total	201	167	146	111	52	184	71	39	10	11	4	42	28	14	4	1084

Table 2 Example of TSCS table of students' page views generated 2-minutes intervals

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	The numb	Column label															
2	Row label	13:02	13:04	13:06	13:08	13:10	13:12	13:14	13:16	13:18	13:20	13:22	13:24	13:26	13:28	13:30	Total
3	Students	9	7					2	1				1				20
4	Students	4	1	1	4	1	10	4	1				1				27
5	Students	8	4				3	2					1				18
6	Students	4	3	5			3	1	1				1				18
7	Students	5		2	2	4		1	1								15
8	Students	6	1	1	3		2	1	1	1				1			17
9	Students	6	4	6			2	1	1					1			21
10	Students			8	6			1	1				1				17
11	Students	4	6	2	3	4	4	4	1		1			1			30
12	Students	6	3	1	6		5	1	1				1				24
13	Students		5	2	4		1	3	1				1	1			18
14	Students		4	3	2	11	1	1	1				1				24
15	Students		2	4	5	5	2	2	1				1				22
16	Students		5	2	7		10	1	1				1				27
17	Students	5	6					2	1				1				15
18	Students	5	2	8			8						3	1			27
19	Students	6	5				2	1	2				2				18
20	Students	6	2		3		3							5			19
21	Students		5	16	17		8	2	2				2				52
22	Students	5	6	2			7	3	2				1				26
23	Students	6	6	1		1	3	1	1					1			20
24	Students					11	2	1						3	2	2	21
25						... More students' records cut here ...												

4. DISCUSSION AND CONCLUSION

Processing of the macro is completed in several seconds. About using the macro during class, the application of the macro to produce a TSCS table for an actual class reveals that such table can be generated without any problems. Depending on the manner by which teacher proceeds with a lesson, however, certain cases have not enough time to use the macro. If students are asked to perform lesson related tasks, such as computing practice, a teacher can run the macro more than once. Aside from enabling teachers to understand the transitions that underlie students' page views, a TSCS table for course items also provide data on variations in students' levels of concentration (Table 1). In the classroom, the teacher manipulated the computer at the teacher's desk and displayed the course materials on the projector. Therefore, the number of students viewing the course materials is smaller because they were looking at the projector screen while listening to the teacher's instruction; i.e. they received the lesson without opening the course materials on their own PC.

Note that certain risks are associated with the use of the TSCS tables. The TSCS table has undeniable possibility of looking at the downloaded materials. Furthermore, after a teacher provides directions on opening a course material, students spend about 1 to 2 minutes accessing the resource. The aforementioned issues should be considered before teachers advance to the next lesson. TSCS tables reflect the viewing behaviors of students over time as transitioning numerical values. During class, teachers can use the tables to visualize the responses of students to instructions. Additionally, the tables provide information regarding which student access teaching materials without following a teacher's instructions and those who exhibit a delay in opening the materials (Table 2). These learners can be distinguished on the basis of transitioning numerical data.

5. ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number 15K00498.

6. REFERENCES

[1] Dierenfeld, H. and Merceron, A. 2012. Learning Analytics with Excel Pivot Tables. In *Proceedings of the 1st Moodle Research Conference* (MRC2012). Retalis, S. and Dougiamas, M. (Eds), 115-121.

[2] Google Analytics. 2016. http://www.google.com/analytics/

[3] Mazza, R. and Dimitrova, V. 2005. Generation of graphical representations of student tracking data in course management systems. In *Proceedings of the 9th International Conference on Information Visualization*. London, UK, July 6-8, 2005.

[4] Mazza, R. and Milani, C. 2004. Gismo: a graphical interactive student monitoring tool for course management systems. *International Conference on Technology Enhanced Learning*. Milan, 1-8.

[5] Konstantinidis, A. and Grafton, C. 2013. Using Excel Macros to Analyses Moodle Logs. In: *2nd Moodle Research Conference* (MRC2013). (4th and 5th October, 2013, Sousse, Tunisia).

[6] Romero, C., Ventura, S. and Garcia, E. 2008. Data mining in course management systems: Moodle case study and tutorial. *Computers and Education*, 51 (1), 368–384. DOI= http://dx.doi.org/10.1016/j.compedu.2007.05.016

[7] Zhang, H. and Almeroth, K. 2010. Moodog: Tracking Student Activity in Online Course Management Systems. *Journal of Interactive Learning Research*. 21(3), 407-429.

Predicting STEM Achievement with Learning Management System Data: Prediction Modeling and a Test of an Early Warning System

Michelle M. Dominguez
Dept. of Educational Psychology & Higher Education, University of Nevada, Las Vegas
4505 S Maryland Parkway #3003
Las Vegas, NV 89154, USA
+001(702)895-3253
doming90@unlv.nevada.edu

Matthew L. Bernacki
Dept. of Educational Psychology & Higher Education, University of Nevada, Las Vegas
4505 S Maryland Parkway #3003
Las Vegas, NV 89154, USA
+001(702)895-4013
matt.bernacki@unlv.edu

P. Merlin Uesbeck
Dept. of Computer Science, University of Nevada, Las Vegas
4505 S Maryland Parkway #3003
Las Vegas, NV 89154, USA
uesbeck@unlv.nevada.edu

ABSTRACT
Learning management systems log users' behaviors, which can be used to predict achievement in a course. This paper examines the implications of data representations (e.g., dichotomous vs. count vs. principled, per learning theory) and applies forward selection algorithms to predict achievement in a biology course. Accuracy is compared across models. The paper closes with a description of an ongoing experiment that employs the prediction model, tests how multiple versions of an early alert message impact students' access of learning resources, and compares the influence of messaging approaches related to personalization and feedback.

Keywords
Learning Management Systems, Prediction Modeling, Early Warning Systems, STEM learning, Learning Theory

1. INTRODUCTION
In response to issues with student performance, retention, progression, and completion [5], universities and educational software providers are developing "early warning systems" to identify students likely to obtain poor outcomes [3]. This paper explores whether logs of students' use of course content can inform models that predict these students' performance. Further, if models can be developed that rely on only behaviors occurring in the earliest weeks of a semester [1], intervention activities can be initiated in time to help students prevent negative outcomes [2].

Undergraduate students utilize a learning management system (LMS) for multiple functions. Based on design features of LMS resources, patterns of student activity may implicate how to represent data in prediction models [4]. For instance, it is more appropriate to model use of a downloadable file as a dichotomous event that should impact learning if it occurs once (indicating that a student has obtained the file) compared to zero times (indicating the student has not). In contrast, resources designed for repeated use online, such as practice quizzes, are best captured as count data. We examine implications of different representations of LMS resource use on the accuracy of prediction models, examine whether the most accuracy model predicts performance in subsequent samples, and whether the model can provide a basis for alerting students about their potential for poor achievement.

2. METHODS

2.1 Participants
For the development of the prediction model, LMS logs capturing behavioral data were gathered for 326 students of an Anatomy and Physiology course at a large, public university in the U.S. Of those sampled, 73% were female and 36% were from underrepresented minority groups. To examine the application of the prediction model on future students, additional samples of 298 and 349 students were drawn from the subsequent Spring and following Fall semesters. All three semesters employed an identical syllabus, an analogous schedule through the observation period, and a cloned set of LMS-hosted materials.

2.2 Materials
Prediction modeling used machine data extracted from server logs of users' behavior-based activity in the LMS from the first four weeks of the course (i.e., prior to any exam). Early warning could then be generated and sent in time for learners to adjust tactics or seek help prior to their first unit exam (i.e., in Week 5). The logs were aggregated and enriched using Splunk [7], a platform for search and modeling of machine data, and tables of metadata about content items. Classification of items into resource types was handled by human research programmers. Models were built and evaluated in RapidMiner [6].

2.3 Procedure
The course that provided data was a traditional large lecture class with a companion site on the LMS, Blackboard Learn. Students could access course materials at any time from the start of the semester, and all use was optional. The frequency and timing of each resource access was recorded and coded by a unique item identifier and time stamp. To represent planful, timely, and recurring use of content items, counts of accesses were captured on a weekly basis. Total use was captured per week and for the four-week period. Behavioral data were merged with performance data. The final grade served as the outcome label. Grades were converted to a binary outcome reflecting students' success (1) or failure (0) to earn a grade of 80%, the minimum "B" score needed to earn credit for STEM majors. Data were parsed into tabular form, enriched, and pivoted into counts per week per student in Splunk. Forward Selection, Weka logistic regression algorithms employing Leave-One-Out cross validation were produced for the models, which were evaluated for accuracy (e.g., **κ**, recall).

2.4 Model Estimation and Application
Four versions of the data were generated. The first version included the *count* of times a student accessed each content item. The second version treated all data as *dichotomously* used or not used in a period. The third version included *both* count of logs and the dichotomous versions of the data. The final version was a *principled* model guided by learning theory and awareness of instructional design intentions of the instructor; a dichotomous

representation was used for items that could be used only once (i.e. the download of a notes document) and count representations for resources that should provide benefits when used repeatedly (e.g., accessing a quiz to repeatedly self-test).

Based on the Kappa (κ) statistic and supplemented with recall metric (i.e., critical for identifying those predicted to struggle), the most accurate model produced during the test phase was then applied to the subsequent two semesters of the same biology course. Content names and date ranges of access were aligned and all potential attributes, as both dichotomous and count, were transformed using the prediction model equation to calculate *z*-values for all students, which was then converted to probability. A probability greater than 0.5 corresponded to passing with a B or better and a probability less than 0.5 corresponded to C or worse.

3. RESULTS & DISCUSSION

Differences in prediction accuracy appear in Table 1. Representing the data as only count or dichotomous produced models with accuracy better than chance, (κ = .161 and κ = .165, respectively). The model with data as both count and dichotomous improved the accuracy to κ = .224, however the recall of students to be targeted by the early warning system (i.e., those who fail to obtain a B or Better) fell. Compared to the metrics obtained by the first three models, the model employing principled representation produced the best combined accuracy, κ = .212; recall = 84.24%. It appears that drawing inferences from LMS design features and learning theory to make data representation choices maximizes the predictive accuracy of a model. We next tested its subsequent utility for identifying students at risk of poor outcomes.

3.1 Application of Prediction Models to Subsequent Samples

Using the most accurate model (Principled, Table 1), attributes and weights were applied to the new data sets to generate predictions. Kappa decreased to .071 compared to training and testing phase (κ = .212). Recall achieved with spring data was 85.14%, on par with recall obtained with the training (84.24%). This model accurately identified more than 4 of 5 future biology students who would eventually fail to earn a B. Of those labeled, half did obtain a B or Better (precision = 51.85%, initial principled model precision was 63.01%). This level of accuracy is sufficient to warrant consideration of the model for utilization in an early warning system as it is high enough to provide accurate warnings to students at risk of a poor outcome.

4. ONGOING RESEARCH

4.1 Implementation of Early Warning Systems

A follow-up study is currently underway to examine the application of the prediction model in an early alert system and whether issuing an alert to students could change student behavior or achievement. The principled version of the data model was programmed into Splunk in order to calculate the likelihood the students (N = 430) in the current semester would obtain a B or better. An early warning message was sent from the instructor through the LMS correspondence tool. Each message included a salutation, indication of the upcoming exam, and a redirect of the student to helpful resources available on the LMS for students to use (i.e., advice from A or B-earners from prior semesters, about tactics used; modules training students to apply these tactics). The students were randomly assigned to 8 groups, which included varying combinations of the message to test the importance of personalizing the message and framing with feedback. The message was sent Monday of Week 5, four days before their exam.

4.2 Preliminary Findings

Of the 326 students that were messaged, 26.4% accessed the Advice page within 24 hours after receiving the message. In total, 37.4% of the messaged students accessed the Advice page before the exam later that week. Effects on motivation, behavior, and achievement will be analyzed when available.

Table 1. Prediction models using different versions of data and using best model on subsequent semesters

Data representation	κ	Accuracy (%)	Precision (%)	Recall (%)	True: Predicted			
					1:1	1:0	0:1	0:0
count	.16	61	61	82	48	94	34	150
dichotomous	.17	60	63	72	63	79	52	132
both	.22	63	65	73	69	73	49	135
principled	.21	63	63	84	51	91	29	155
Future Semesters								
Spring	.07	53	52	85	33	117	22	126
Fall	.15	58	57	81	56	112	34	147

Note. The baseline for test data versions (count, dichotomous, both & principled) is 56%. The baseline for the Spring use data is 51% and the baseline for Fall use data is 52%.

5. ACKNOWLEDGMENTS

This project was supported by National Science Foundation Award number DRL-1420491, university sponsorship and UNLV Information Technology.

6. REFERENCES

[1] Baker, R., Lindrum, D., Lindrum, M.J., Perkowski, D. (2015) Analyzing Early At-Risk Factors in Higher Education e-Learning Courses. Proceedings of the 8th International Conference on Educational Data Mining, 150-155

[2] Hernandez, P. R., Schultz, P., Estrada, M., Woodcock, A., & Chance, R. C. (2013). Sustaining optimal motivation: A longitudinal analysis of interventions to broaden participation of underrepresented students in STEM. Journal of educational psychology, 105(1), 89.

[3] Jayaprakash, S. M., Moody, E. W., Lauría, E. J., Regan, J. R., & Baron, J. D. (2014). Early alert of academically at-risk students: An open source analytics initiative. Journal of Learning Analytics, 1(1), 6-47.

[4] Macfadyen, L. P., & Dawson, S. (2010). Mining LMS data to develop an "early warning system" for educators: A proof of concept. Computers & Education, 54(2), 588-599.

[5] Nandeshwar, A., Menzies, T., & Nelson, A. (2011). Learning patterns of university student retention. Expert Systems with Applications, 38(12), 14984-14996.

[6] Rapidminer [Computer software]. (2015). Retrieved from http://www.rapidminer.com

[7] Splunk [Computer software]. (2015). Retrieved from http://www.splunk.com.

Comparison of Selection Criteria for Multi-Feature Hierarchical Activity Mining in Open Ended Learning Environments

Yi Dong
Institute for Software Integrated Systems
Vanderbilt University
1025 16th Ave S, Nashville, TN 37212
yi.dong@vanderbilt.edu

John S. Kinnebrew
BRIDJ
283 Newbury St, Boston, MA 02115
john.kinnebrew@gmail.com

Gautam Biswas
Institute for Software Integrated Systems
Vanderbilt University
1025 16th Ave S, Nashville, TN 37212
gautam.biswas@vanderbilt.edu

ABSTRACT

This paper extends our previous work on a Multi-Feature Hierarchical Sequential PAttern Mining (MFH-SPAM) algorithm for deriving students' behavior patterns from their activity logs in an Open-Ended Learning Environment(OELE). The new algorithm is computationally efficient, and we compare the results generated by the two algorithms.

1. INTRODUCTION

Open-Ended Learning Environments [2, 5] present students with a challenging problem-solving task, along with information resources and tools for solving the task. The complexity of the learning task drives a need for dynamic and adaptive scaffolding to help novice students become effective learners. Learner models and formative assessments need to include representations that capture students' problem-solving processes in addition to their knowledge and performance in the task domain. The wealth of data that can be collected from computer-based environments provides opportunities for developing algorithms to accurately model, understand, assess students' learning behaviors and strategies.

In past work, we have developed a hierarchical sequence mining methods [3] for assessing and comparing students' learning strategies and behaviors from their interaction traces collected from OELEs. We then applied a classifier wrapper method [4] to discover smaller subsets of mined patterns that better differentiate students behavior patterns between two groups of students [7]. To address the computational complexity problems with this method while retaining the advantages of the hierarchical approach, this paper applies another selection criteria: Information Gain [6] to derive differential patterns. We conduct experimental studies to analyze student behaviors and compare the two methods.

2. BACKGROUND: MFH-SPAM

Sequential PAttern Mining (Sequential PAttern Mining) algorithm performs a Depth First Search (DFS) traversal to find all possible patterns that exceed a pre-defined frequency threshold from a data set that contains sequences of item sets [1]. SPAM employs a bitmap representation for the patterns and data sequences, which makes it easy to (1) derive pattern extensions and (2) find pattern matches in data sequences during traversal. The DFS search proceeds by extending action sequences with (1)*Sequence-extension* step (*S-step*), which extends pattern by adding a new itemset to the **end** of current pattern sequence, and (2)*itemset-extension* step (*I-step*), which adds a new item to the **last** itemset of a current sequence as an extension.

The MFH-SPAM algorithm further extends the original SPAM algorithm by adding two steps: (1) the *hierarchical-extension* step (*H-step*), which provides a way to get into more details for given actions by bringing in hierarchical representations, and (2) the *feature-extension* step (*F-step*) which makes patterns more informative by associating features with corresponding actions [7]. As a result, MFH-SPAM finds many more patterns compared to the SPAM algorithm. MFH-SPAM also allows for gaps between items(actions) that make up a pattern [3] to accommodate noise tolerance in the action sequences.

In general, even for reasonably-sized domains, the basic MFH-SPAM algorithm returns thousands of patterns, and this presents challenges in extracting the more important patterns that best characterize and differentiate student behavior. Given the computational complexities of the classifier-wrapper method used earlier [7], this paper develops a new selection criterion based on information gain [6] to identify activity patterns that distinguishes students based on their pre- to post-test learning gains measured outside of the system. The information gain for a given pattern P_1 is computed from the reduction in *Shannon entropy* when P_1 becomes known, where *Shannon entropy* for a sample data is a measure of its homogeneity. We focus on analyzing patterns with high information gain that are good differentiators between student groups.

3. CASE STUDY AND RESULTS

We run our case study on a dataset that was generated from an experiment we ran with 98 middle school students who used a learning by teaching environment, *Betty's Brain*, in a science class for a period of approximately six weeks. Learner are tasked to construct a correct causal map of a science process by reading resources, and use the knowledge learned to construct and assess the correctness of their causal map during the study. In one of our current study, students worked on a thermoregulation unit.

The students' learning gains from pre- to post-test provided us with two equally distributed groups: 49 high performers in Group 1, and 49 low performers in Group 2. We then ran the two versions of the MFH-SPAM algorithm: (1) with the classifier wrapper method, and (2) with the information gain methods to select the top 10 patterns that best differentiate the two groups. The results, presented in Tables 1 and 2 respectively, list the mean frequency of usage and the standard deviation for each selected pattern.

Table 1: Classifier Wrapper method.

Pattern	Mean(STD)	Mean(STD)
	High Group	Low Group
editlink;quiztaken	25.9(21.9)	10.6(13.3)
editmap-eff-sup	24.1(17.6)	12.3(11.5)
quiz;editmap	14.0(18.5)	7.5(12.7)
editmap-eff;quiz;expl	11.2(9.7)	5.9(8.6)
quiz;editlink;read	6.1(7.6)	2.3(2.5)
read-shrt;read;editmap;linkadd	4.0(3.3)	2.4(1.7)
read-long	19.8(30.2)	34.0(29.2)
read-shrt;editlink	13.8(9.1)	19.7(10.1)
editmap;quizview	6.8(5.6)	9.7(10.9)
editmap-ineff-unsup;read	5.6(5.1)	8.5(6.4)

Table 2: Patterns with High Information Gain

Pattern	Mean(STD)	Mean(STD)
	High Group	Low Group
quiz	95.3(51.2)	72.9(51.1)
expl	90.4(75.8)	70.0(68.9)
editlink;quiztaken	25.9(21.9)	10.6(13.3)
editmap-eff-sup	24.1(17.6)	12.3(11.5)
editmap-ineff;quiz	20.3(16.3)	14.6(12.7)
editlink;quiz;editmap	16.7(21.4)	7.2(16.1)
quiz;editmap;read	6.4(7.7)	2.8(2.9)
quiz;editlink;read	6.1(7.6)	2.3(2.5)
read-long	19.8(30.2)	34.0(29.2)
take-notes	9.5(11.3)	23.9(24.4)

Both methods find patterns that are good differentiators between the two groups of students. For example, *read-long* (sufficiently long duration read actions) has a high to low performer use ratio of 1 : 2. On the other hand, the quiz followed by an edit link followed by a resource read (*quiz;editlink;read*) has a high to low performer use ratio of 2.75 : 1. Another pattern *editlink;quiztaken* (high to low performer use ratio of 2.5 : 1) found by both methods indicates high performers are better able to use the quizzes(*quiztaken*) to check the correctness of their maps, and to direct their information seeking activities. The classifier wrapper method applying cross validation where decision tree is built multiple times for each chosen pattern, results in larger amount of calculations for information gain, whereas our new method which theoretically finds patterns with highest information gain based on i-frequency, calculates information gain only once for each pattern. Moreover, the new method tends to find shorter patterns because that shorter patterns occupying fewer bits in action sequences for i-frequency based information gain calculation, have lower value of pattern entropy which lead to higher information gain compare to longer patterns with similar usage ratio [6].

4. DISCUSSION AND CONCLUSIONS

In this paper, we have extended an initial version of MFH-SPAM by developing additional selection criteria for pattern selection and also allowing for gaps in the pattern generation from action sequences. The new method is computationally efficient than the previous approach(running time reduced from 28 seconds to 16 seconds) while retaining the strength of finding frequent patterns that are good differentiators.

In future work, we will perform more systematic analysis of the differences between groups using hypothesis testing methods. In addition, we will use correlational analysis to study in more depth the relations between behaviors and performance. We will also work toward using the patterns derived to detect learner behaviors online, and develop scaffolding and hint mechanisms that combine behavior and performance analysis to help students become better learners in OELEs.

5. ACKNOWLEDGMENTS

This work is supported by NSF IIS grant number # 1548499.

6. REFERENCES

[1] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu. Sequential pattern mining using a bitmap representation. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 429–435. ACM, 2002.

[2] G. Clarebout, J. Elen, W. L. Johnson, and E. Shaw. Animated pedagogical agents: An opportunity to be grasped? *Journal of Educational multimedia and hypermedia*, 11(3):267–286, 2002.

[3] J. S. Kinnebrew, K. M. Loretz, and G. Biswas. A contextualized, differential sequence mining method to derive students' learning behavior patterns. *Journal of Educational Data Mining*, 5(1), 2013.

[4] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artif. Intell.*, 97(1-2):273–324, Dec. 1997.

[5] S. Land. Cognitive requirements for learning with open-ended learning environments. *Educational Technology Research and Development*, 48(3):61–78, 2000.

[6] J. R. Quinlan. Induction of decision trees. *Mach. Learn.*, 1(1):81–106, Mar. 1986.

[7] C. Ye, J. S. Kinnebrew, J. R. Segedy, and G. Biswas. Learning behavior characterization with multi-feature, hierarchical activity sequences. In *8th International Conference on Educational Data Mining*, June 2005.

A Data-Driven Framework of Modeling Skill Combinations for Deeper Knowledge Tracing

Yun Huang
Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA, USA
yuh43@pitt.edu

Julio D. Guerra-Hollstein
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA, USA
jdg60@pitt.edu

Peter Brusilovsky
School of Information Sciences
University of Pittsburgh
Pittsburgh, PA, USA
peterb@pitt.edu

ABSTRACT

This paper explores the problem of modeling student knowledge in complex learning activities where multiple skills are required at the same time and combinations of skills might carry extra specific knowledge. We argue that in such cases mastery should be asserted only when a student can fluently apply skills in combination with other skills. We propose a data-driven framework to model skill combinations for tracing students' deeper knowledge, and also propose a novel evaluation framework which primarily focuses on the mastery inference quality. Our experiments on two real-world datasets show that proposed model significantly increases mastery inference accuracy and more reasonably distributes students' efforts comparing with traditional Knowledge Tracing models and its non-hierarchical counterparts.

Keywords

complex skill, multiple skill, composition effect, robust learning, deep learning, Knowledge Tracing, Bayesian Network

1. INTRODUCTION

Knowledge Tracing (KT) [2] has been established as an efficient approach to model student skill acquisition in intelligent tutoring systems. The essence of this approach is to decompose overall domain knowledge into elementary skills and map each step's performance to the knowledge level of a single skill. However, KT assumes skill independence in problems that involve multiple skills, and it is not always clear how to decompose overall domain knowledge. Recent research demonstrated that there is additional knowledge related to specific skill combinations; in other words, the knowledge about a set of skills is greater than the "sum" of the knowledge of individual skills [6], some skill must be integrated (or connected) with other skills to produce behavior [9]. For example, students were found to be significantly worse at translating two-step algebra story problems into expressions (e.g., 800-40x) than they were at translating two closely matched one-step problems (with answers 800-y and 40x) [6]. In particular, research on computer science education has long argued that knowledge of a programming language cannot be reduced to a sum of knowledge about different constructs since there are many stable combinations (patterns, schemas, or plans) that have to be taught. We present a data-driven framework for modeling skill combinations and evaluating student models for adaptive tutoring in order to achieve deeper knowledge tracing.

2. PROPOSED FRAMEWORK

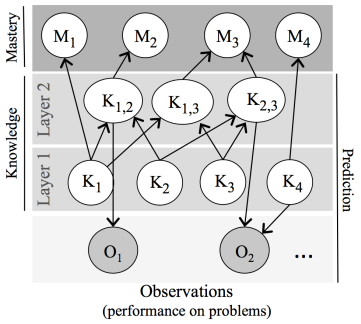


Figure 1: The Bayesian network structure of CKM-HSC.

We construct a Bayesian network called *conjunctive knowledge modeling with hierarchical skill combinations (CKM-HSC)* with the following knowledge structure:

I The first layer consists of basic individual skills (e.g., K_1) that capture the basic understanding of each skill.

II The intermediate layers consist of skill combinations (e.g., $K_{1,2}$), which can be derived from smaller skill units that capture a deeper knowledge level of each individual skill. Now, we consider only skill combinations from two basic individual skills.

III The last layer consists of *Mastery* nodes (e.g., M_1) for each individual skill, which reflects the idea of granting a skill's mastery based on relevant skill combinations' knowledge levels. Now, we compute the joint probability of each relevant skill combination being known as the probability of the current skill being mastered.

To learn the network structure, we propose a greedy search algorithm where a pre-ordering of the skill combination candidates is given as input, and during each iteration, the data likelihood of the network incorporating a new skill combination is compared to that of the optimal network so far. We now replace the search procedure with an empirical thresholding method, which generates an almost identical network with much less time. It selects combinations based on the following criteria: 1) the difficulty difference between the combined skill and its hardest individual one should be positive and large; 2) the difficulty of the combined skills should be high; 3) an item with higher difficulty should be more likely to require combined skills; and 4) each item can only have a limited number of skill combinations. To perform a dynamic knowledge estimation, we use the roll-up mechanism, as in [1]. For performance prediction, we apply Noisy-and gates on item nodes (e.g., O_1) as in [1, 3].

Table 1: Dataset descriptive statistics.

Dataset	#obs.	#items	#skills	avg #skills/item	#users	%correct
SQL	17,197	45	34	5 (from 1 to 10)	366	58%
Java	25,988	45	56	5 (from 1 to 11)	347	67%

To address the limitation of predictive performance metrics [7, 5], we propose a multifaceted data-driven evaluation framework that includes mastery accuracy and effort, the item discriminative index [3], and performance prediction metrics. The basic idea of the mastery accuracy metric is that once a student model asserts mastery for an item's required skills, the student should be unlikely to fail the current item. Meanwhile, the mastery effort metric empirically quantifies the number of practices that are needed to reach a level of mastery for a given set of skills. These metrics extend our recent learner effort-outcome paradigm [5] and Polygon multifaceted evaluation framework [7].

3. STUDIES

We used datasets collected from SQL and Java programming learning systems from 2013 to 2015 at the University of Pittsburgh. Table 1 shows the descriptive statistics (with multiple attempts). We conducted a 10-fold student stratified cross-validation. For each metric, we reported the average value across 10 folds and with a 95% confidence interval, based on the t-distribution. We used the Bayes Net Toolbox to construct all the models. On average, we extracted 14 and 30 skill combinations on SQL and Java datasets.

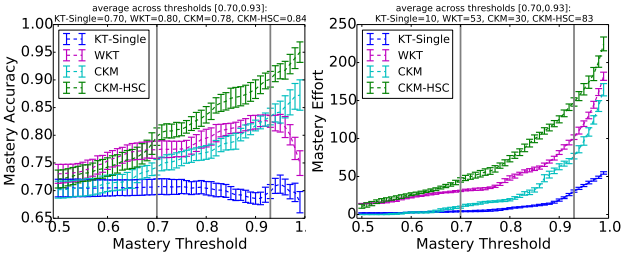


Figure 2: Mastery accuracy and effort comparison on Java dataset. Grey lines denote regions with enough data points to compute mastery metrics and with high enough values to be considered as proper mastery thresholds.

Our first study investigates whether the proposed skill combination incorporated model is better than traditional KT models. We compare classic Knowledge Tracing (KT-Single) [2], Weakest Knowledge Tracing (WKT) [4], and our proposed conjunctive knowledge modeling without (CKM) or with skill combinations (CKM-HSC) (Figure 2). On both datasets, CKM-HSC has a comparable predictive performance to other models, but it has significantly better mastery accuracy than other models. Although it requires more efforts to reach mastery, we think that such "extra" practices is necessary for reaching an acceptable mastery inference accuracy. We further conduct a drill-down analysis for mastery effort by splitting skills into two groups based on whether they involve skill combinations. We find out that for skills that involve skill combinations, WKT would blindly distribute students' efforts among different application contexts, risk students reaching mastery by practicing simple problems, and also guide students to spending more efforts on skills without combinations. On the other hand, CKM-HSC saves students' efforts on basic individual skill understanding and on skills without skill combinations. It requires students to focus more on applying skills in different contexts combined with other skills. We further conduct two studies demonstrating that using a hierarchical structure is better than using a flat independent structure for incorporating skill combinations, and that our modeling can be improved by adding external knowledge (such as expert knowledge or skill combinations' textual proximity) for skill combination extraction. Details are reported in [8].

4. CONCLUSIONS

Our work serves as a first attempt to consider the skill application context for modeling deeper knowledge in a student model using data-driven techniques. We also propose a novel data-driven evaluation framework for such complex skill student models. We only consider pairwise skill combinations as the skill application context; it will be to interesting to consider more complex skill combinations. Such combinations should have a natural connection with the concept of *chunk* in cognitive psychology for defining expertise. Meanwhile, to address the problem of computational complexity we now employ some heuristics. We should explore alternative approaches and more efficient techniques. We will also consider working with larger datasets and datasets with more sparse connections among variables. We expect that our model can provide more benefits when deployed in real-world tutoring systems. For example, it might enable better remediation and raise students' awareness of pursuing true mastery.

5. REFERENCES

[1] C. Conati, A. Gertner, and K. Vanlehn. Using bayesian networks to manage uncertainty in student modeling. *User Modeling and User-Adapted Interaction*, 12(4):371–417, 2002.

[2] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1995.

[3] J. De La Torre. An empirically based method of q-matrix validation for the dina model: Development and applications. *Journal of educational measurement*, 45(4):343–362, 2008.

[4] Y. Gong, J. E. Beck, and N. T. Heffernan. Comparing knowledge tracing and performance factor analysis by using multiple model fitting procedures. In *Proc. 10th Int. Conf. Intelligent Tutoring Systems*, pages 35–44. Springer, 2010.

[5] J. P. González-Brenes and Y. Huang. Your model is predictive but is it useful? theoretical and empirical considerations of a new paradigm for adaptive tutoring evaluation. In *Proc. 8th Intl. Conf. Educational Data Mining*, pages 187–194, 2015.

[6] N. T. Heffernan and K. R. Koedinger. The composition effect in symbolizing: The role of symbol production vs. text comprehension. In *Proc. 19th Annual Conf. Cognitive Science Society*, pages 307–312.

[7] Y. Huang, J. P. González-Brenes, R. Kumar, and P. Brusilovsky. A framework for multifaceted evaluation of student models. In *Proc. 8th Int. Conf. Educational Data Mining*, pages 203–210, 2015.

[8] Y. Huang, J. Guerra, and P. Brusilovsky. Modeling skill combination patterns for deeper knowledge tracing. In *6th Int. Workshop on Personalization Approaches in Learning Environments (In Submission)*, 2016.

[9] K. R. Koedinger, A. T. Corbett, and C. Perfetti. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive Science*, 36(5):757–798, 2012.

Generating Semantic Concept Map for MOOCs

Zhuoxuan Jiang[1], Peng Li[1], Yan Zhang[2], Xiaoming Li[1]
School of Electronics Engineering and Computer Science
Peking University, Beijing, China
[1]{jzhx,lipengcomeon,lxm}@pku.edu.cn, [2]zhy@cis.pku.edu.cn

ABSTRACT

The task of re-organizing the teaching materials to generate concept maps for MOOCs is significant to improve the experience of learning process, e.g. adaptive learning. This paper introduces a novel and tailored Semantic Concept Map (SCM), and we design a two-phase approach based on machine learning methods to generate it.

1. INTRODUCTION

With the increasing development of Massive Open Online Courses (MOOCs) in recent years, it is believed that how to efficiently re-organize the course materials to serve for better learning is worthy of discussion [6].

In the traditional computer-assisted education, concept map is useful but usually involves domain experts. Considering the large amount of MOOCs, an information system that behaves like an expert and provides the skeleton of a concept map can be more effective.

Unlike partially organized e-textbooks, we can not directly identify concepts from various MOOC materials merely through stylistic features, so machine learning based method is leveraged. Moreover, in order to reduce the cost of labelling, semi-supervised framework is adopted in this paper. Rather than generating various relationships between concepts, we define a novel Semantic Concept Map (SCM) which considers semantic similarity as the only relationship without regard to complex and hierarchy ones. Due to its concision and universality, this map can be applied widely to more courses. Figure 1 shows the two-phase approach including 1) concept extraction and 2) relationship establishment.

2. RELATED WORK

Plenty of work about automatically constructing concept maps has been studied with data mining techniques, such as association-rule mining, text mining and specific algorithms [7]. However, these methods are designed for either specific data sources or special learning settings. Due to the diversity of MOOCs settings, they can hardly be leveraged here.

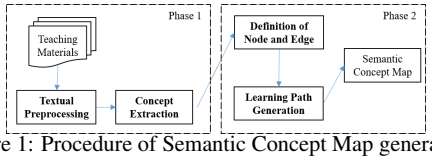


Figure 1: Procedure of Semantic Concept Map generation.

The task of terminology extraction in computer science field is similar to our machine learning based concept extraction [1], but those methods mainly concern about proper nouns or named entity recognition (NER) for generating knowledge graph [5]. Actually this kind of task is corpus-dependent.

3. GENERATING SEMANTIC CONCEPT MAP

Semantic Concept Map. SCM is composed of entities and edges. Formally, denote $SCM = \{C, R\}$ where $C = \{c_1, c_2, ..., c_n\}$ is a set of concepts. Each concept c_i is denoted by a terminology (including phrase), and unique in C. $R = \{r_{11}, r_{12}, ..., r_{ij}, ..., r_{nn}\}$ is a set of relationships between concepts. Each weight value r_{ij} means the degree of semantic similarity between c_i and c_j. The key steps shown in Figure 1 are following.

1.Textual Preprocessing. This step includes tokenization, filtering stop words and removing code and html tags, as well as word segment for Chinese if necessary. We also conduct conflation. All data are randomly shuffled before being learnt and tested, which is partially equivalent to cross-fold validation.

2.Concept Extraction. We leverage CRF+semi-supervised framework to solve this task as a problem of sequence annotation [2]. The labels needed to be predicted of each word are defined as three categories: B, I and O, which respectively mean the beginning word of a concept, the internal word of a concept and not a concept. Feature definition is a key part of machine learning method. Then we design the course- and instructor-agnostic features to meet the diverse materials including stylistic, structural, contextual, semantic and dictionary features. In order to reduce the heavy cost of human labeling, the idea of self-training is leveraged when training data [3].

3.Definition of Node and Edge. The weights of nodes could have different definitions. For example, the more frequent a concept is present in the lecture notes, the more fundamental it is. So the metric of term frequency (tf) can be defined as the node weights, named for *fundamentality*. The diverse teaching materials put together are partitioned to documents corresponding to each video. Moreover, low-frequency concepts may be the key ones of each corresponding unit. So we can define the second metric, Term Frequency and

Table 1: Performance of different concept extraction methods.

	Precision	*Recall*	*F1*
TF@500	0.402	0.500	0.446
TF@1000	0.600	0.746	0.665
BT	0.099	0.627	0.171
SC-CRF	**0.890**	**0.842**	**0.865**
SSC-CRF	0.875	0.783	0.826

Inverted Document Frequency (*tfidf*) which is ideal for quantifying the importance of a concept. As to the weights of edges, the Cosine distance of two word vectors of concepts are defined as the semantic similarity, because the word vectors learnt by word2vec have a natural trait that semantically similar vectors are close in the Cosine space and vise verse [4].

4.Learning Path Generation. The learning path depends on the definition of node and edge in the last step. For example in terms of importance, starting from some concept, each time we choose top k most semantically similar concepts and regard the most important one within the top k as the next node of the path. When choosing the subset of top k candidates, we also consider their locational order of first appearance in the lecture notes.

4. EXPERIMENTS

We collect the teaching materials of an interdisciplinary course conducted on Coursera, including lecture notes (video transcripts), PPTs, questions. The instructors and two TAs help label the data.

We select several baselines to extract concepts from MOOCs materials for comparison. The preprocessing is identical for baselines.

- **Term Frequency (TF)**: This is a statistic baseline.
- **Bootstraping (BT)**: A rule-based iterative algorithm given several patterns which contain true concepts.
- **Supervised Concept-CRF (SC-CRF)**: A supervised CRF with all features but semi-supervised algorithm.

Table 1 shows the performance between baselines and our approach (SSC-CRF). The results also show the necessity of machine learning based methods. Figure 2 manifests that semi-supervised learning is competitive with supervised learning. But considering only half labor consumed, semi-supervised learning is feasible and necessary.

Based on the definitions of node and edge mentioned before, the two kinds of SCMs generated look like Figure 3. Starting from the most fundamental concept, *Node*, the first five successors on the path are: *Edge* → *Element* → *Set* → *Alternative* → *Vote*, which are from basic concepts to advanced ones. Starting from the most important concept, *PageRank*, the first five successors on the path are: *PageRankAlgorithm* → *SmallWorld* → *Balance* → *NashBalance* → *StructuralBalance*. We can see they are not only important along with the course syllabus, but also semantically similar.

5. CONCLUSION

In this paper we mainly propose an approach to re-organize existing teaching materials to generate a novelly-defined SCM for facilitating the learning process in MOOCs. This work is a promising start for content-based adaptive learning since hierarchical and multiple relationships of a complete concept map can be incrementally replenished, and meanwhile this map can be extended to more courses and domains. Experiments show a good efficacy of the semi-supervised

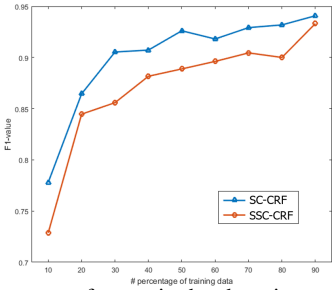
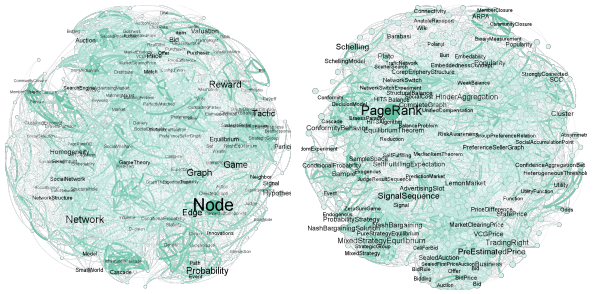


Figure 2: Performance of supervised and semi-supervised learning.



(a) For fundamentality (b) For importance

Figure 3: Two kinds of Semantic Concept Map.

machine learning algorithm and the CRF framework. And the learning paths defined based on SCMs can be humanly modified further to satisfy the requirements of different learners. In future work SCM could be utilized for generating course Wiki via crowdsourcing, hinting concept in forum discussions, etc. Large-scale student knowledge tracing in MOOCs is also doable by associating concepts with questions. Moreover, methods of transfer learning and deep learning may be more effective to extract the abstract concepts from multiple courses and diverse materials.

6. ACKNOWLEDGMENTS

This research is supported by NSFC with Grant No.61532001 and No.61472013, and MOE-RCOE with Grant No.2016ZD201.

7. REFERENCES

[1] K. S. Hasan and V. Ng. Automatic keyphrase extraction: A survey of the state of the art. In *ACL*, pages 1262–1273, 2014.

[2] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML'01*, pages 282–289, 2001.

[3] A. Liu, G. Jun, and J. Ghosh. A self-training approach to cost sensitive uncertainty sampling. *Machine Learning*, 76(2-3):257–270, 2009.

[4] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space, 2013. `http://arxiv.org/abs/1301.3781`.

[5] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich. A review of relational machine learning for knowledge graphs, 2015. `http://arxiv.org/abs/1503.00759v3`.

[6] Z. A. Pardos, Y. Bergner, D. T. Seaton, and D. E. Pritchard. Adapting bayesian knowledge tracing to a massive open online course in edx. In *EDM'13*, pages 137–144, 2013.

[7] C. Romero and S. Ventura. Educational Data Mining: A Review of the State of the Art. *IEEE Trans. on Systems, Man, and Cybernetics*, 40(6):601–618, 2010.

How to Judge Learning on Online Learning: Minimum Learning Judgment System

Jaechoon Jo
Dept. of Computer Science and Engineering
Korea University
Seoul, South Korea
(+82)2-3290-2684
jaechoon@korea.ac.kr

Heuiseok Lim
Dept. of Computer Science and Engineering
Korea University
Seoul, South Korea
(+82)2-3290-2684
limhseok@korea.ac.kr

ABSTRACT
The popularity of online education environment is growing due to the Massive Open Online Course (MOOC) movement. Many types of research in educational data mining (EDM) and Learning Analytics have focused on solving assessment challenges; however, the large number of students enrolled in MOOCs makes it difficult to assess learning outcomes. Thus, it is necessary to develop an automatic learning judgment system. In this study, we designed and developed a minimum learning judgment system that assesses minimal learning using a word game performance measure. In the system, a student watches a video containing educational content and is subsequently tested on information retention by playing a word game that tests the student on the video content. This learning judgment system tests minimal learning of educational content without requiring significant effort from either the instructor or the student. We conducted experiments to show a performance of the system and the result shows about 95% (Pass judgment: 95.1%, Fail judgment: 94.8%) performance.

Keywords
MOOC, Flipped Learning, Judge System, Online Education, Data Collection, Educational Data Mining.

1. INTRODUCTION
Over 10 million people participate in online learning courses, which has resulted in the proliferation of the use of MOOCs. Consequently, the number of online courses that implement online learning platforms, such as Moodle, Coursera, and edX has steadily increased in online education. Online learning platforms provide useful learning data for learner modeling and learning analysis. Learning data provide various types of information that can assess student participation in online courses, such as the number of logins, the number of postings made to discussion boards, and various types of learning outcomes [1]. However, due to the high number of students participating in MOOCs, one critical problem that must be addressed is how instructors can conduct learning assessments that determine learning. Traditional assessment methods are not suitable for online education. Most existing most online learning platforms require a simple quiz and online exam based on traditional assessment methods [2]. Many quizzes and exams can be a burden to both instructors and students. Thus, it is necessary to develop an automatic learning judgment system that can quickly and simply assess learning.

In this paper, we aim to design and develop a minimum learning judgment system. Our approach aims to solve learning assessment challenges in online education in order to minimize the amount of effort required by teachers and learners in assessing learning. Anyone can access and utilize this system [1] at no cost for the purposes of conducting research and collecting educational data. We will present the overall system process and the experiments that were conducted to test the system.

2. MINIMUM LEARNING JUDGMENT
In this paper, we define minimum learning as a behavior state of initial learning, which is automatically determined after a student watches a video and is assessed using a recognition process that measures the frequency effect theory of words used in the video content [3]. In other words, watching video content is the minimal behavior of learning apart from understanding. It does not mean that system can assess understanding of content knowledge.

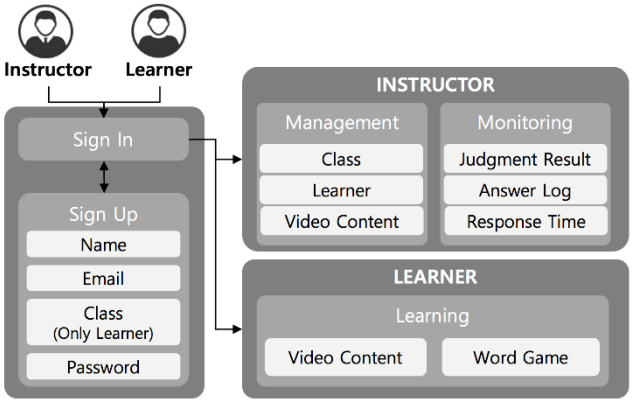


Figure 1. Overall System Process

Figure 1 presents the overall system process for users. After registering an instructor, the instructor can add classes and upload video contents. Words are extracted from the uploaded video content and word frequency is automatically calculated. After registering a learner with a class, the student can learn by viewing video content that the instructor has uploaded. After viewing the video, the student can begin the word game. In the word game, the student decides whether words did or did not appear in the video. The system judges minimum learning by measuring the student's response time and accuracy in the word game. Finally, the

[1] Minimum Learning Judgment System: http://www.mljs.org

instructor checks the minimum learning results, the word game logs and a response time for each word.

The words that appear in the word game use word frequency from uploaded video content and the Sejong corpus (made by www.sejong.or.kr). In order to select words for the word game, words are selected by measuring the weight of each word, which is based on both previous videos that the student learned and on the current video content that student is watching. Each student plays a word game with a different word set in which different weights correspond to different learning logs. The weight of a word is calculated as follow:

$$w_{ij} = tf_{ij} \times log\left(\frac{N}{n}\right) + 1 \quad (1)$$

A weight $w_{ij} > 0$ is associated with each word i in a video content j. Let tf_{ij} refer to the frequency of word i in video content j. Let N refer to the number of video contents viewed by the student in the entire set of video contents. Let n be the number of video contents where w_{ij} appears in N.

In total, 14 words are selected for the word game. The seven highest-frequency words are selected from video content and seven words that have the same word length as the video content words are selected randomly from the Sejong corpus. These latter seven words that do not appear in video content will referred to as "noisy words." The reasoning behind choosing seven words is that the video content is based on short-term memory (STM) [4]. When the word appears, the student chooses the word within two seconds. According to language cognition theory, cognition time of a known word takes between from 700ms to 1200ms [3]. Taking into consideration the conditions that may affect the speed of web environment networks, this system adds and subtracts 500ms to the recorded response time.

3. EXPERIMENTS

3.1 Participants and Analysis

In order to get a criteria score, we conducted an experiment in which we tested 60 undergraduate students. Thirty-two of the students were male, 28 of the students were Female, and the ages of the selected participants ranged from 19 to 27. Each participant viewed video content and then played the word game. Then, participants' attention levels were assessed on a five-point scale using the Likert-type scale. The data collected from the system was analyzed based on the expectation-maximization (EM) algorithm using WEKA. Table 1 presents the results of our analysis.

Table 1. Result of Clustering

Cluster	A	B
Attention	1.004 (SD. 0.027)	3.6084 (SD. 1.0678)
Score	6.0588 (SD. 2.0694)	9.5569 (SD. 2.566)

Cluster A refers to the set of participants who did not pay attention while watching the video content. On average, the members in Group A selected six of the 14 words correctly. Cluster B refers to the set of participants who paid attention while watching the video content. On average, the members in Group B selected 9 of the 14 words correctly. Therefore, the criteria for the minimum learning judgment system correspond to seven correctly selected words.

3.2 Test and Results

Finally, we ran a minimum learning assessment to determine whether learners watched the video content or not. In a test set, 240 undergraduate students participated in the experiment. Participants were divided into two groups: an experiment group, which consisted of 120 students who watched the video content, (Pass) and a control group, which consisted of 120 students who did not watch the video content (Fail). Table 2 presents the results of the test, which measured precision and recall.

Table 2. Result of Test Table

		Real				
		Pass	**Fail**	**Precision**	**Recall**	**F1**
System	**Pass**	118	10	92.1875	98.3333	95.1
	Fail	2	110	98.2142	91.6666	94.8

For the Passing group, the result of minimum learning judgment demonstrated a precision rate of 92% and a recall rate of 98%. For the Failing group, the result of minimum learning judgment demonstrated a precision rate of 98% and a recall rate 91%. Finally, the performance of system shows about 95%.

4. CONCLUSIONS AND FUTURE WORK

This paper presents how a minimum learning judgment system can solve assessment challenges in online education environments by reducing the work required by both instructors and learners. This system shows about 95% performance but it is optimized for the training data set. Thus, we need to conduct further experiments and analyses using machine learning algorithms and educational data mining technologies in order to develop and strengthen our system. Finally, we hope **this system can be utilized by instructors and researchers** for their educational and research purposes.

5. ACKNOWLEDGMENTS

This work was supported by the ICT R&D program of MSIP/IITP. [2016, Development of distribution and diffusion service technology through individual and collective Intelligence to digital contents].

6. REFERENCES

[1] Fazel Keshtkar, Jordan Cowart, and Ben Kingen 2015. Analyzing Students' Interaction Based on their Responses to Determine Learning Outcomes. In *Proceedings of the 8th International Conference on Educational Data Mining, Poster and Demo Papers*, 588-589.

[2] Shumin Jing 2015. Automatic Grading of Short Answers for MOOC via Semi-supervised Document Clustering. In *Proceedings of the 8th International Conference on Educational Data Mining, Poster and Demo Papers*, 554-555.

[3] Jaechoon Jo, and Heuiseok Lim 2015. A Development of Minimum Learning Judgement System of Online Learner, *Korean Association of Computer Education*, 20, 1, 63-66.

[4] Randall, W. Engle 2002. Working Memory Capacity as Executive Attention, *Current Directions in Psychological Science*, 11, 1 (Feb. 2002), 19-23

Guiding Students towards Frequent High-Utility Paths in an Ill-Defined Domain

Igor Jugo, Božidar Kovačić, Vanja Slavuj
Department of Informatics
University of Rijeka
Croatia
+385 51 584 711
ijugo@inf.uniri.hr bkovacic@inf.uniri.hr vslavuj@inf.uniri.hr

ABSTRACT
This paper presents an exploratory data mining methodology for discovering frequent high-utility learning paths from a database of student interactions with an adaptable tutoring system. The discovered paths are used to present recommendations to students in order to make the learning process more efficient. The novelty of our approach is twofold: a) the process of data preparation, path evaluation and path discovery is completely autonomous; and b) the process is executed on a growing dataset of learning traces while the students are advancing through the knowledge domain. We present the system overview and the obtained results.

Keywords
Sequential pattern mining, computer-based learning environment, high-utility patterns, recommendations.

1. INTRODUCTION
The objective of a tutoring system is to guide each student towards a predefined goal such as completing a lesson, task, or mastering a skill. Guiding students is more complex in ill-defined domains [4] where it is not possible to break down the learning units into single skill tasks, and the students have the freedom to choose/create their own path through the domain. One such web-based system has been developed at our institution to serve as an additional learning platform in a blended learning approach applied in a number of courses. The process presented in this paper is the third and final part (the first two being: 1) a communication layer that enables the system to communicate with DM tools, and 2) a clustering method [3] that discovers groups of students that use the system in a similar manner) of a new infrastructure developed with the goal of improving the adaptivity of our system [2].

While attempting to master the knowledge domain presented in our system, each student creates a large number of learning paths. Most of the students will need multiple interactions with a unit until it is mastered/completed, e.g., after a failed attempt they realize they need to learn some other (lower-level) units and then they come back to complete the first unit. The objective of the system is to offer recommendations to students about which unit to select (when the student is just starting a session or a new learning "run") or which unit to learn next (right after finishing learning a unit). For this, we need to discover productive frequent paths leading to, and following after, each unit. To discriminate between productive and unproductive frequent patterns we decided to construct a new dataset based on the database of learning paths and then feed that dataset into a high-utility sequential pattern mining algorithm USPAN [5] which requires a sequence database that contains both the unit IDs and their "profit" (in our case – the calculated efficiency of each path).

2. DISCOVERING FREQUENT HIGH-UTILITY PATHS
The system supports two types of learning activities:

a) LEARNING - presenting learning materials followed by a question about the unit, and initial questions about the connected underlying units (units below in the domain structure created by teacher). If the student answers all the questions correctly, the path is considered optimal and the change in the student's overlay model is calculated. If the student offers an incorrect answer to a question about a connected unit, the system will transfer the student to learn that unit, and the whole process is recursively repeated. Therefore, one learning "run" can consist of a number of learning units and a number of questions answered;

b) REPETITION - answering a series of questions about a unit without presenting learning materials. A visualization of four possible paths for a sample domain consisting of five units (A-E) is presented in Figure 1.

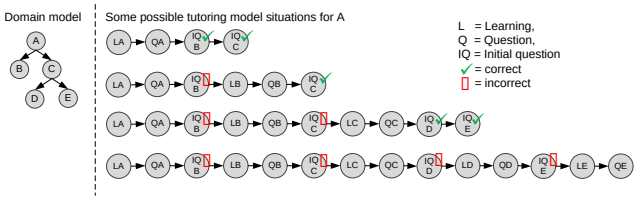


Figure 1. Possible variations in learning path lengths

The basic components for profit calculation, based on four paths presented in Figure 1, are presented in Table 2. Each unit has a set threshold value *t* that the student has to reach (by answering the questions). The current value of *t* for each unit in the domain model represents the student's model.

Table 1. Path profit calculation

UNITS									
A	**B**	**C**	**D**	**E**	**∑**	**PL**	**Ls**	**Qs**	**IQs**
t=10	*t*=10	*t*=6	*t*=8	*t*=10					
+2	+0.2	+0.2			2.4	4	1	1	2
+3	-0.1 +2	+0.2			5.1	6	2	2	2
+2	-0.1 +2	-0.1 +2	+0.2	+0.2	6.2	10	3	3	4
+1	-0.1 +1	-0.1 +1	-0.1 +2	-0.1 +2	6.6	14	5	5	4

The following expression is used to calculate "profit" *P* of path *x*:

$$P_x = \left(\frac{\sum_{i=1}^{Units} cq_i/t_i}{Q} + \frac{\sum_{i=1}^{Units} ciq_i/t_i}{IQ} \right) * \frac{PL_{min}}{PL}$$

We summarize the changes *c* that followed from answering a question about each of the units (Units) occurring in *x*, divided by the unit threshold value *t*. This accounts for the difficulty of the presented questions. The sum is then divided by the total number of questions answered (Q). The same is done for initial questions (IQ). Finally, the total change is multiplied by the difference between minimal and actual path length (PL). This penalizes longer paths as they are caused by incorrect answers to initial questions. Minimum path length (PL_{min}) is calculated based on the number of units added to the learning structure at the time the learning activity took place. The tutoring model determines the number of items in the learning structure based on the student's overlay model state, e.g., according to Figure 1, if the student starts the LEARNING activity with unit A, having previously completed units B and C, the tutoring module will not add any units to the learning structure (except for A, making $PL_{min} = 1$).

After the learning traces of all the students that are using the knowledge domain have been evaluated, they can be transformed into a sequence database for the USPAN algorithm. The transformation algorithm creates two databases for each unit in the domain – a set of paths consisting of units learned before the current unit ("prefix") and a set of paths consisting of units learned after ("suffix"). Each transformed sequence has a maximum length of 6 units. Both datasets are then converted to the correct format of the USPAN algorithm implementation in SPMF [1]. The system is now ready to discover high-utility frequent paths (HUFP). We run the algorithm on each dataset under the condition that a unit has been learned by at least five students, i.e., we must have a minimum of five paths in the dataset, although there can be much more if the students have been struggling with the unit. When the process is complete, all the discovered high-utility frequent paths are written to the database. Once the system has updated the HUF paths database the recommendation selection algorithm chooses the unit to be recommended at the beginning and the end of each learning activity. The algorithm considers: a) the student model; b) whether the unit has already been recommended and/or followed by this student; and c) which recommendation was most followed by other students.

3. RESULTS

We tested our system in two different knowledge domains, with 31 and 69 learning units, used by 30 and 20 students, respectively. The results are presented in Table 2. The "D.SET" column contains the number of learning traces in the system at the time the process was executed. The number of HUFPs discovered at each execution (divided by "prefix" and "suffix" paths) is presented in the next three columns. Column "UNITS" presents the number of units for which HUFPs were discovered. As expected, the unique number of units reached the total number of units in the domain in last two executions. The number of recommendations presented to students and the unique number of units for which the recommendations were presented are displayed in the next two columns.

Table 2. Results for first domain

No	D.SET	HUFPs	PRE	SUF	UNITS	REC.	UN. REC	FOL.
1	1206	21	5	16	7	12	5	5
2	1616	35	20	15	15	83	24	39
3	2068	115	65	50	22	204	15	36
4	2504	121	79	42	18	225	15	12
5	2912	89	52	37	21	47	12	13
6	3314	418	227	191	31	417	23	35
7	3604	538	289	249	31	332	24	22

Finally, the last column presents the number of recommendations followed (clicked) by students. The percentage of followed versus total number of recommendations varied from 5 to 47 percent. Further analysis will be performed to evaluate the overall impact of the recommendation mechanism on the learning process.

4. CONCLUSION

The presented methodology was implemented in a web-based ITS and tested on two different domains. We believe that the main improvements to the system can be made in: a) the interaction-to-path transformation algorithm, by implementing additional logic to recognize branch/level changes in the domain hierarchy which can reflect student's strategy, and b) the recommendation selection algorithm, by implementing additional logic to minimize repetition and optimize the selection process.

5. ACKNOWLEDGMENTS

This research is a part of the Project "Enhancing the efficiency of an e-learning system based on data mining", code: 13.13.1.2.02., funded by the University of Rijeka, Croatia.

6. REFERENCES

[1] Fournier-Viger, P. et al. 2014. SPMF: a Java Open-Source Pattern Mining Library. *Journal of Machine Learning Research*, 15, 3389-3393.

[2] Jugo, I., Kovačić B. and Slavuj, V. 2016. Increasing the Adaptivity of an Intelligent Tutoring System with Educational Data Mining: a System Overview, in *Int. Journal of Emerging Technologies in Learning*, 11, 3.

[3] Jugo, I., Kovačić, B. and Tijan, E.. 2015. Cluster analysis of student activity in a web-based intelligent tutoring system, in *Pomorstvo: journal of maritime studies*, 29, 80-88.

[4] Lynch, C. et al. 2006. Defining Ill-Defined Domains; A literature survey. *In ITS2006: Proc. of Intelligent Tutoring Systems Ill-Defined Domains Workshop*, Taiwan, 1-10.

[5] Yin, J., Zheng, Z., & Cao, L. 2012. Uspan: an efficient algorithm for mining high utility sequential patterns. In *18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 660-668.

Computing Pointers Into Instructional Videos

[Extended Abstract] *

Andrew Lamb
Stanford University
andrew.lamb@stanford.edu

Jose Hernandez
Stanford University
josehdz@stanford.edu

Jeffrey Ullman
Stanford University
ullman@cs.stanford.edu

Andreas Paepcke
Stanford University
paepcke@cs.stanford.edu

ABSTRACT

We examine algorithms for creating indexes into ordered series of instructional lecture video transcripts. The goal is for students and industry practitioners to use the indexes towards review or reference. Lecture videos differ from often-examined document collections such as newspaper articles in that the transcript ordering generally reflects pedagogical intent. One challenge is therefore to identify where a concept is *primarily* introduced, and where the resulting index should thus direct students. The typically applied TF-IDF approach gets tricked in this context by artifacts such as worked examples whose associated vocabulary may dominate a lecture, but should not be included in a good index. We contrast the TF-IDF approach with algorithms that consult Wikipedia documents to vouch for term importance. This method helps filter the harmful artifacts. We measure the algorithms against three human-created indexes over the 90 lecture videos of a popular database course. We found that *(i)* humans have low inter-rater reliability, whether they are experts in the field or not, and that *(ii)* one of the examined algorithms approaches the inter-rater reliability with humans.

1. INTRODUCTION

Lecture videos of online classes are clumsy when students wish to review course materials. It is impossible to access just a particular portion of interest. A solution would be an automatically created index similar to the reference at the end of a book. The facility would allow access into portion of videos where a particular topic is discussed.

We compared several algorithms that create such an index for every course video. Raw material are the closed caption files that are often available for educational video. Those files contain transcripts of the audio, paired with timing information at roughly sentence granularity.

We paid three humans with varying domain expertise to carefully index the video transcripts from a Stanford online database course. We compared the three resulting indexes to each other, and to results from the algorithms. We make the three reference indexes and the database course video caption files available to the public in hope of eliciting indexing approaches beyond those that we explored.

*A full version of this paper is available at `http://ilpubs.stanford.edu:8090/1140/1/indexer.pdf`

2. EXPERIMENTS

Our first experiment took a traditional approach, selecting words for the index that appeared disproportionately often in certain lectures (TF-IDF [1]). We then incorporated lexical information, by only considering phrases that followed certain part-of-speech patterns. Finally, we introduced external knowledge from Wikipedia into an algorithm's indexing decisions. Note that none of the algorithms included supervised learning, as we do not assume the existence of a training set for all courses. The following subsections introduce the algorithm (families) beyond the TF-IDF version.

2.1 Leveraging Linguistic Information

The first algorithm tags parts of speech in the lecture transcripts. It then extracts as index candidates phrases that consist of adjectives followed by one or more nouns. For example, "equality condition" or "XML data" would be included.

2.2 Adding External Knowledge

Note that phrases gain importance because of both their role in a document but also from their semantic meaning in the broader world. Variants of our next algorithms therefore integrate Wikipedia as a knowledge source.

2.2.1 Boosting Documents

The first variant concatenates to each lecture a closely related Wikipedia page, and then uses the techniques of Section 2.1 to choose phrases for the index. For example, lecture title "View Modifications Using Triggers", yields as the first Wikipedia result a page titled "Database trigger." This page is appended to the lecture transcript. Using either n-grams or adjective-noun phrases as candidate keywords, the algorithm chooses phrases with TF-IDF over the combined document for the index.

2.2.2 Boosting Phrases

This algorithm first creates a list of candidate index terms using adjective-noun phrases. These candidates are ranked by their TF-IDF score summed over **all** Wikipedia documents.

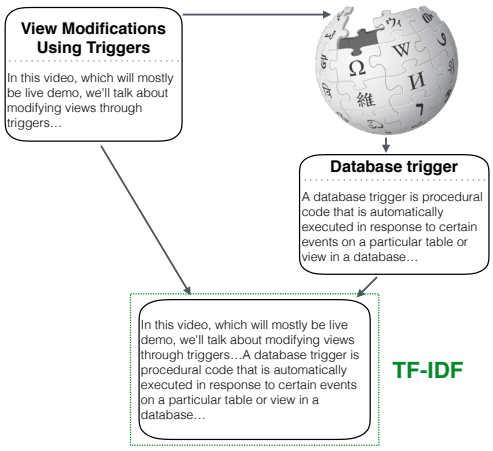


Figure 1: The Document Boosting algorithm searches for a Wikipedia page using the title of the lecture, concatenates the result to the lecture, and then runs TF-IDF over the combined document.

Next, this global candidate ranking is combined with a basic TF-IDF approach to form a final score that combines global knowledge (from Wikipedia) with local knowledge (from the specific lecture video).

We also experimented with only boosting phrases of at least two words, based on the intution that longer phrases are often meaningful, but appear infrequently and are therefore given low scores by TF-IDF. We call this alternative "Phrase Boosting N-Grams" in Figure 3.

Rank	Phrase
1	**view**
2	**materialized view**
3	materialized
4	**query**
5	**view query**
6	**virtual view**
7	**modify**
8	user query
9	**base table**
10	**modify command**
11	**index**
12	insert command
13	multivalued dependency
14	**database design**
15	user

Figure 2: The top 15 keywords from 'Materialized Views' by Phrase Boosting with N-grams. Phrases that also appear in the gold index are marked in bold.

2.3 Results

We evaluated each algorithm by computing Cohen's Kappa agreement between the algorithm and a gold set created by unifying two of the human indexes[1]. We chose a widely employed inter-rater reliability measure because indexing is highly subjective. Given this absence of absolute truth we therefore treated the algorithms as we would have measured reliability of an additional human indexer.

[1]One of the human indexes was excluded because it sometimes included words that did not appear in the lecture.

Kappa values do not have a universally agreed upon interpretation, but values in the range we observe (about 0.15 to 0.3) have been interpreted as indicating "slight" to "fair" agreement. We measured agreement of 0.325 between the humans in the gold index. This value is therefore the measure to beat.

Algorithm	κ
TF-IDF	0.205
TF-IDF with Adjective-Noun Chunks	0.079
Document Boosting	0.209
Document Boosting with Adjective-Noun Chunks	0.142
Phrase Boosting	0.204
Phrase Boosting N-Grams	**0.237**

Figure 3

The metrics for all of the algorithms are shown in Figure 3. The Phrase Boosting N-Grams algorithm, which favors longer words, performed best with a Cohen's Kappa of 0.237. The Document Boosting algorithm is able to slightly improve on TF-IDF, by filtering superfluous keywords using the external knowledge from Wikipedia. Note that Cohen's Kappa can sometimes be problematic when using an unbalanced dataset. In our full paper, we evaluate the algorithms with complementary metrics to guard against potential pathological cases.

Figure 2 shows the set of keywords extracted from a lecture on 'Materialized Views' by the Phrase Boosting with N-grams algorithm, in Figure 2 . Of the top 15 keywords marked by the algorithm, 11 were included in the gold index marked by humans (for this lecture there were 18 keywords in the gold set), and the algorithm produces a ranking that is similar to the humans. Of the keywords ranked highly by the algorithm that were not in the gold index, some ('materialized', 'insert command', 'multivalued dependency') are relevant to the course, but perhaps not essential to the specific lecture. The last two keywords, 'user' and 'user query' expose a weakness of the algorithm, where it is difficult to discern phrases that are used frequently, but not essential to the lecture concept.

3. CONCLUSION

We started to tackle the task of choosing the most important phrases from a collection of lectures, to construct a random-access index analogous to those in the back of books. Going forward we will use this capability to construct student support facilities such as automatically answering learner questions with references to relevant lecture clips, and recommendation tasks, such as finding the best study materials given a student's progress through a course.

4. REFERENCES

[1] C. D. Manning, P. Raghavan, and H. Schutze. *Introduction to Information Retrieval*. Cambridge University Press, 32 Avenue of the Americas, New York, NY 10013-2473, USA, 2008.

Hierarchical Cluster Analysis Heatmaps and Pattern Analysis: An Approach for Visualizing Learning Management System Interaction Data

Ji Eun Lee
Utah State University
jieun.lee@aggiemail.usu.edu

Mimi Recker
Utah State University
mimi.recker@usu.edu

Alex J. Bowers
Columbia University
bowers@tc.columbia.edu

Min Yuan
University of Utah
min.yuan@eccles.utah.edu

ABSTRACT
This paper presents a form of visual data analytics to help examine and understand how patterns of student activity – automatically recorded as they interact with course materials while using a Learning Management System (LMS) – are related to their learning outcomes. In particular, we apply a data mining and pattern visualization methodology in which usage patterns are clustered using hierarchical cluster analysis (HCA) then visualized using heatmaps to produce what is called a clustergram. We illustrate the application of this methodology by building two clustergrams in order to explore university students' LMS activity patterns using both semester and weekly summary data. The resulting clustergrams reveal differences in LMS usage between high-achieving and low-achieving/dropout students.

Keywords
Hierarchical Cluster Analysis, Heatmap, Learning Management System, Visual Data Analytics

1. INTRODUCTION
With the explosive growth in the use of LMS to support instructional activities, several recent studies have applied Educational Data Mining (EDM) to analyze the vast datasets collected by LMS. Results from such studies can help identify at-risk learners, monitor student performance, and inform course re-design [4, 5].

Some approaches tend to take a variable-centered approach, examining features and trends in key usage variables. In contrast, a person-centered approach can highlight individual sub-groups of students that share common data patterns [1, 7], that when pattern analyzed, link to important differences in overall course or educational outcomes. In this way, data points are not aggregated, thereby obscuring their individual patterns [6].

This study takes the latter, person-centered approach. As a form of visual data analytics, we describe and apply a data mining and pattern visualization methodology, in which usage patterns are clustered using hierarchical cluster analysis (HCA) then visualized using heatmaps to produce what is called a *clustergram* [1]. We illustrate the application of this methodology by analyzing data collected from a widely used LMS, Canvas. In particular, we address two questions: To what extent do clustergrams help understand patterns of student activity in the course? How do these patterns of activity relate to student learning outcomes?

2. BACKGROUND
2.1 EDM and LMS
Much prior EDM research applied to LMS data has typically taken a variable-centered approach by examining usage at an aggregated level [3]. While useful, these results aggregate and average users' behaviors, and thus make it difficult to recognize the diverse patterns displayed by different groups of users [6]. Thus, in the present study, we take a more person-centered approach to visually investigate what sub-groups of students may share common patterns, and how these relate to their learning outcomes.

2.2 Hierarchical Cluster Analysis Heatmaps
HCA is a multivariate statistical method for classifying related units in an analysis across high dimensionality data. More recently, HCA has been combined with heatmap visualizations, called a *clustergram* [1]. The clustergrams represent each participant's row of data across each of the columns of variables as a color block, using stronger intensities of one color to represent lower levels of the variable, and increasing intensities of a different color to represent higher levels. We apply cluster analysis heatmap visualizations to Canvas LMS data from a large, online course. In this way, we test the utility of the analysis and visualization technique when applied to the potentially larger data patterning and visualization issues around these types of student interaction data.

3. METHODS AND DATA SOURCES
The data are drawn from a larger dataset containing all student recorded by the Canvas LMS at a medium-sized U.S. western university. For the present study, we extracted the student interaction data from a large (N=139) introductory level mathematics online course taught during the fall 2014 semester.

Two clustergrams were built, one with semester summary data and the other with weekly summary data. First, for the clustergram using the semester summary data, all student activity data were transformed to z-scores in order to standardize variance. HCA was applied to cluster both rows and columns. Color gradients ranges from colder blue for -3 SD below the mean to a hotter red for value +3 SD above the mean. Second, for the clustergrams using the weekly summary data, we used raw data and HCA was applied to only the rows. In addition, we applied k-means clustering on the rows for more precise interpretation of clustergrams. Lastly, student final course grade was included as an overall outcome variable in the final column. For all analyses, we used the R studio with the "ComplexHeatmap" packages. Regarding algorithms, the clustergrams were clustered using K-means, then HCA (using average linkage and Euclidean distance) was applied to each row-cluster.

4. RESULTS
4.1 Clustergram using semester summary
Figure 1 presents a section of the clustergram using the semester summary data. As shown in Figure 1, most students with lower

activity (Cluster 1) either received a grade of F or withdrew (W) from the course, whereas many students with higher activity (Cluster 3) received a grade of A.

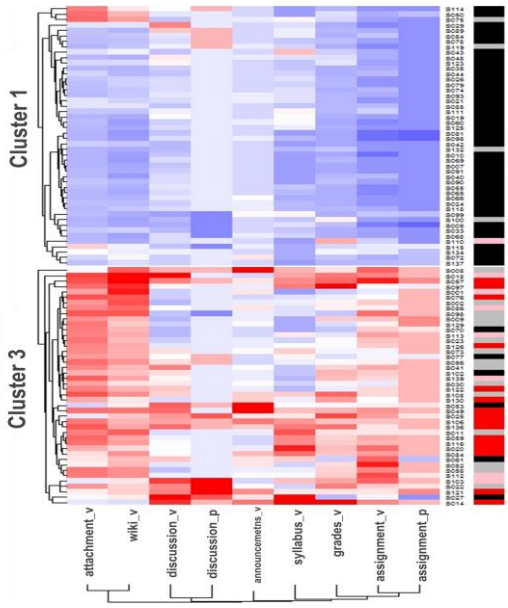


Figure 1. Section of the clustergram using the semester summary data (for full image: goo.gl/Y7VFHJ)

A correlational analysis revealed that the variables related to engaging with 'assignment' features had the highest positive correlations with final grades (r = .70, p < .05). Variables related to views of grades (r = .56, p < .05), wiki (r = .42, p < .05), syllabus (r = .35, p < .05), and attachments (r = .35, p < .05) had the next highest positive correlations with final grades. The remaining variables (views of announcements, participation in discussions) were not significantly correlated with final grades.

4.2 Clustergram using weekly summary

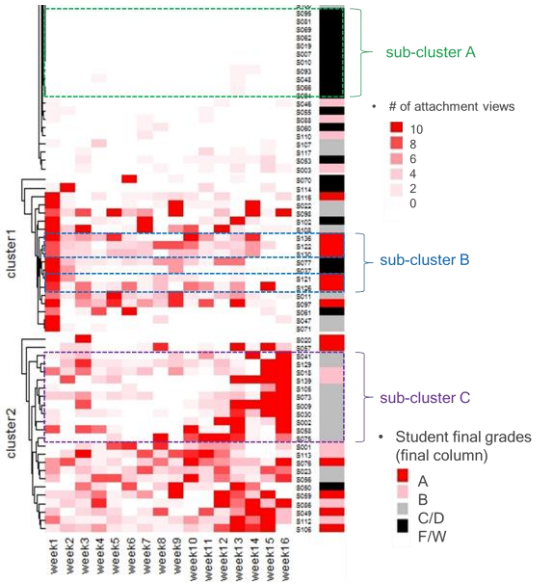


Figure 2. Section of the clustergram for the number of students' attachment views by week (for full image: goo.gl/IwcCch)

In order to investigate how student activities changed over the course of the semester, we built a clustergram using the weekly summary data. Figure 2 presents a section of the clustergram for the number of students' 'attachment' views by week.

The clustergram shows that the students with a grade of A (sub-cluster B) showed relatively consistent views of attachments over the course of the semester. Interestingly, the students with a grade of A (sub-cluster B) tended to show higher attachment views at the beginning of the course and more consistently throughout the semester. However, the students with grades of C/D (sub-cluster C) tended to have higher attachment views at the end of the course, representing perhaps a less-successful 'cramming' strategy.

5. CONCLUSION

This study demonstrates the utility of cluster analysis heatmap visualizations as a means to use visual data analytics to examine student patterns of activity at different grain sizes (week vs. semester). Combining this technique with the large sets of LMS provides a unique opportunity to examine the patterns of student activity as they relate to overall student outcomes. This type of visual data analytics expands the number of tools available for instructors and administrators to help identify the features and specific LMS interaction data that are most useful to their students. As recent critiques of LMS interaction data have shown that past analytic methods are insufficient to understand the rich complexity of how students learn through an LMS [2], this study provides an additional means to approach these complex data analytic issues.

6. REFERENCES

[1] Bowers, A.J. 2010. Analyzing the longitudinal K-12 grading histories of entire cohorts of students: Grades, data driven decision making, dropping out and hierarchical cluster analysis. *Practical Assessment, Research & Evaluation.* 15, 7, 1-18.

[2] Koedinger, K. R., D'Mello, S., McLaughlin, E. A., Pardos, Z. A., and Rosé, C. P. 2015. Data mining and education. *Wiley Interdisciplinary Reviews: Cognitive Science.* 6, 4 (Jul. 2015), 333-353.

[3] Macfadyen, L. P., and Dawson, S. 2010. Mining LMS data to develop an "early warning system" for educators: A proof of concept. *Computers & Education*. 54, 2 (Feb.2010), 588-599.

[4] Picciano, A. G. 2014. Big data and learning analytics in blended learning environments: Benefits and concerns. *International Journal of Artificial Intelligence and Interactive Multimedia*. 2, 7 (Sep. 2014), 35-43.

[5] Romero, C., Ventura, S., and García, E. 2008. Data mining in course management systems: Moodle case study and tutorial. *Computers & Education*. 51, 1 (Aug. 2008), 368-384.

[6] Wilkinson, L., and Friendly, M. 2012. The history of the cluster heat map. *The American Statistician*. 63, *2* (Jan, 2012), 179-184. DOI= http://10.1198/tas.2009.0033

[7] Xu, B., and Recker, M. 2011. Understanding teacher users of a digital library service: A clustering approach. Journal of Educational Data Mining. 3, 1 (Oct. 2011), 1-28.

Understanding Engagement in MOOCs

Qiujie Li
School of Education
University of California, Irvine
Irvine, 92617
qiujiel@uci.edu

Rachel Baker
School of Education
University of California, Irvine
Irvine, 92617
rachelbb@uci.edu

ABSTRACT
Previous studies about engagement in MOOCs has focused primarily on behavioral engagement and less attention has been paid to cognitive engagement. This may lead to incomplete or even incorrect understandings about students experience and learning in MOOCs. In this study, we use number of lectures watched as a proxy for behavioral engagement and number of pauses in lectures watched as a proxy for cognitive engagement. Results show that a large proportion of students who were behaviorally engaged (watching lectures) were not cognitively engaged—they almost never paused the lectures or they paused fewer and fewer times as the course went on. This may indicate that being behaviorally engaged does not necessarily mean being cognitively engaged. In addition, we also found that students' number of pauses in lectures is positively associated with achievement and improves the prediction of achievement.

Keywords
Cognitive engagement, behavioral engagement, MOOCs

1. INTRODUCTION
Engagement in MOOCs is usually measured by whether students complete learning activities or not (e.g. watching lectures and submitting assessments) and low engagement is used as an indicator of "at-risk" students [4]. However, studies of school engagement have proposed that engagement has three components: behavioral engagement, cognitive engagement, and emotional engagement, and that measuring engagement solely as task completion may focus only on behavioral engagement and overlook the multifaceted nature of engagement [1]. To explore the importance of cognitive engagement in MOOCs, this study measured both behavioral engagement and cognitive engagement in MOOC lecture watching to see: 1) whether individuals who were behaviorally engaged were also cognitively engaged, and 2) whether cognitive engagement adds information that is helpful in predicting academic achievement.

1.1 Behavioral engagement
Most of previous studies about engagement in MOOCs have focused on behavioral engagement: participation in academic activities [1]. One of the most commonly used engagement indicators in MOOC studies is participation in lecture watching. For instance, in the most frequently cited paper about engagement patterns in MOOCs, Kizilcec et al (2013) measured student weekly engagement as a function of whether they watched any lecture and submitted any assessment. By using these metrics of task completion, this study inherently conceptualized engagement as behavioral engagement. Similarly, measurements centered around behavioral engagement, such as time spent on lecture resources, have also been used in studies about the relationship between engagement and dropout [4].

1.2 Cognitive engagement
Cognitive engagement refers to the psychological investment in learning and ranges from memorizing to using self-regulated strategies to promote one's understanding [1]. In this study, we measure student's weekly cognitive engagement by how often they paused the lectures they watched (i.e., students stop the lecture while watching it). Some studies about MOOCs have explored the possibility of using video lecture clickstream data, the record of student click events, to measure cognitive engagement [3]. Among all the click events, the pausing event may indicate a higher level of cognitive engagement [3].

2. METHODS

2.1 Sample
This study uses data from one Coursera MOOC, Pre-calculus, offered by University of California, Irvine. It began on October 7th, 2013 and lasted for ten weeks. 50,676 students registered the course and data on 19,548 students who watched at leased one lecture after registration was used in this study.

2.2 Measurement
In this study, weekly behavioral engagement was measured by the number of lectures student watched each week while weekly cognitive engagement was measured by the number of pauses in lectures watched in a given week. In addition, we measured weekly academic achievement in two ways: students' total quiz score (the sum of scores a student got on each quiz he/she attempted each week) and students' average quiz score (the average score on quizzes attempted each week).

2.3 Analysis
We applied a standard clustering technique, K-means, to discover student engagement patterns based on the two measurements to see whether individuals who were behaviorally engaged were also cognitively engaged. We first standardized the engagement score within each week to take into account the difference in participation across weeks and thus to cluster students based on their relative similarity in engagement within each week. Then, we performed the clustering analysis separately for behavioral engagement and cognitive engagement. To get an optimal "goodness of fit" for the data, cluster silhouette, a measure of how similar an individual is to his/her own cluster compared to other clusters, was used to determine the number of clusters. For behavioral engagement, 4 to 9 clusters produced similar cluster

silhouette (above 0.7) and for cognitive engagement, 4 to 8 clusters produced similar cluster silhouette (above 0.6). Accordingly, we performed cluster analysis with all the possible choices. Finally, we chose 4 clusters for both of the two measurements because it gave us enough individuals in each cluster and all the clusters made sense from an educational perspective. In addition, to answer the second research question, we used regression with individual fixed effect to test whether cognitive engagement could predict academic achievement after controlling for behavioral engagement.

3. RESULTS

3.1 Clusters based on different engagement

The four types of behavioral engagement trajectories are: *1)"Strong enders"* (n=157; 0.8%) who watched more lectures than other groups and their average number of lectures watched decreased in the first six weeks but then increased to 50 at the end of the course; *2)"Slow decreasers"* (n=1367; 7.0%) who had a very similar pattern as "stronger enders" except that they kept watching fewer and fewer lectures till the end of the course; *3)"Quick decreasers"*(n=1598; 8.2%) who started at the same place as both "strong enders" and "slow decreasers", but the number decreased at a much faster rate; and *4)"Disengagers"*(n=16426; 84.0%) who watched around 2 lectures in week 1 on average and the number was kept under 1 for the following 9 weeks.

The four types of cognitive engagement trajectories are:*1)"Active stoppers"*(n=41; 0.21%) who, on average, paused each of the lecture they watched more than 10 times in most of the weeks; *2)"Constant stoppers"*(n=367; 1.9%) who, on average, paused each lecture they watched around 5 times in most of the weeks; *3)"Switchers"*(n=1719; 8.8%) who started at the same place as "constant stoppers", but their average number of pauses in lectures watched decreased quickly in the following weeks; and *4)"Continuers"*(n=17421; 89.1%) who almost never paused the lectures they watched or they didn't watch any lectures at all in some of the weeks.

Combining the two types of engagement (see Figure 1), we found that students in clusters with higher levels of behavioral engagement had a larger proportion of individuals who were cognitively engaged. For example, compared with "disengagers" and "quick decreasers", "strong enders" and "slow decreasers" have a smaller percent of "continuers" and larger percent of both "active stoppers" and "constant stoppers". However, being behaviorally engaged does not necessarily mean being cognitively engaged. For example, even though "strong enders" and "slow decreasers" watched the most lectures every week, around 45% them conducted fewer and fewer pauses as the course went on (defined as "switchers") and more than 20% of them almost never paused the lectures (defined as "continuers").

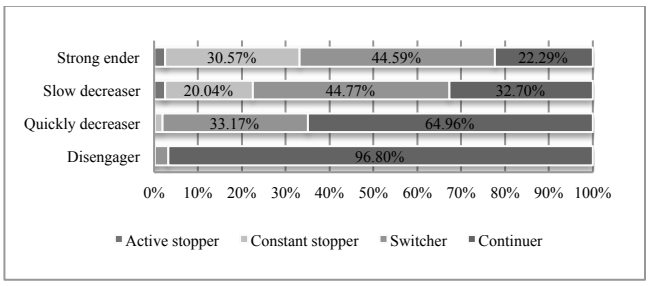


Figure 1. Distribution of cognitive engagement trajectories

3.2 Cognitive engagement and achievement

Using individual fixed effect model (see Table 1), we found that the number of pauses in lectures watched is predictive of both total and average quiz score after controlling for the number of lectures watched. For total quiz score, one more pause is associated with 0.33 points increase in total quiz score and 0.23 points increase in average quiz score. In addition, for both total and average quiz score, the models with the number of pauses in lectures watched fit significantly better than the models that only have number of lectures watched as the predictor. Overall, the results show that our measurement of cognitive engagement is positively associated with achievement and it can make a unique contribution in predicting achievement.

Table 1. Regression of engagement on academic achievement with individual fixed effect

	Total score		Average score	
Number of lectures	0.71^{***}	0.69^{***}	0.04^{***}	0.02^{***}
	(0.004)	(0.005)	(0.001)	(0.001)
Number of pauses per lecture		0.33^{***}		0.23^{***}
		(0.019)		(0.005)
N	79174	79174	79174	79174
R^2	0.281	0.284	0.017	0.054

$^{*}p < 0.05$, $^{**}p < 0.01$, $^{***}p < 0.001$

4. DISCUSSION

Our preliminary results indicate that it is important to take into account cognitive engagement. First of all, using only behavioral engagement may lead to an incomplete or even incorrect understanding about the activeness of students. As we found in this study, some students had relatively high behavioral engagement while decreasing or low cognitive engagement. We may fail to identity some "at-risk" students who visited most of materials but didn't truly engage with the content if we only measure behavioral engagement. In addition, cognitive engagement is found to have its unique contribution in predicting academic achievement and thus can give instructors extra information about student performance in a given course.

5. REFERENCES

[1] Fredricks, J. A., Blumenfeld, P. C., & Paris, A. H. (2004). School engagement: Potential of the concept, state of the evidence. *Review of educational research*, *74*(1), 59-109.

[2] Kizilcec, R. F., Piech, C., & Schneider, E. (2013, April). Deconstructing disengagement: analyzing learner subpopulations in massive open online courses. In *Proceedings of the third international* conference *on learning analytics and knowledge* (pp. 170-179). ACM.

[3] Sinha, T., Jermann, P., Li, N., & Dillenbourg, P. (2014). Your click decides your fate: Inferring information processing and attrition behavior from mooc video clickstream interactions. *arXiv preprint arXiv:1407.7131*.

[4] Taylor, C., Veeramachaneni, K., & O'Reilly, U. M. (2014). Likely to stop? predicting stopout in massive open online courses. *arXiv preprint arXiv:1408.3*

How quickly can wheel spinning be detected?

Noboru Matsuda
Texas A&M University
4232 TAMU
College Station, TX 77843
Noboru.Matsuda@tamu.edu

Sanjay Chandrasekaran
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
sanjayc@andrew.cmu.edu

John Stamper
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
jstamper@cs.cmu.edu

ABSTRACT
We have developed a wheel spinning detector for cognitive tutors that uses a simplified method compared to existing wheel spinning detectors. The detector reads a sequence of the correctness of applying particular skill performed by a student using the cognitive tutor. The response sequence is first fed to Bayesian knowledge tracing to compute a sequence of probability of mastery at each time a skill was applied. The detector uses a neural-network model to make a binary classification for a response sequence into wheel-spinning and none-wheel spinning. To test the accuracy of the detector, we validated the detector using learning interaction data taken from a school study where students used a Geometry cognitive tutor. Human coders manually tagged the data to identify wheel spinning. The results show that the neural-network based detector has high recall (0.79) but relatively low precision (0.25) when combined with Bayesian knowledge tracing that detects mastery cases. The result suggests that the neural-network based detector is practical and has a potential for scalable use such as adaptive online course where cognitive tutors are embedded into online courseware.

Keywords
Wheel spinning; detector; neural network; Intelligent tutoring system; student modeling

1. INTRODUCTION
Cognitive tutors provide mastery learning on cognitive skills [3]. Mastery learning is controlled by a student-modeling technique called knowledge tracing [2] that computes the likelihood of mastering individual cognitive skills to be learned. The output from the knowledge tracer is used to compute an optimal sequence of training problems in such a way a student will achieve the mastery for all cognitive skills quickly [4].

One of the challenges under the paradigm of model-tracing based mastery learning happens when the student model does not detect a mastery within a reasonable amount of time. From the students' point of view, this means that they are continuously posed problems one after another for considerably long time. This phenomenon is called *wheel spinning* that has been coined by Beck and Gong [1].

Wheel spinning, by definition, means a situation in which a student does not reach to a pre-defined mastery level according to the mastery estimation computed by the knowledge-tracing algorithm. Although some students may eventually reach mastery only after working on a considerably many number of problems, it is not practical to assume that students would be persistent under such situation. When students do not see any improvement in their performance and the system merely provide more problems, then they would quickly get frustrated and lose their motivation. It is therefore quite important to detect wheel spinning as soon as possible. A reliable student-modeling technique to predict wheel spinning is there required.

The goal of current study is to develop a detector that detects a risk of wheel-spinning at an early phase of learning in the context of cognitive tutoring. The simplicity and scalability of the technology is one of the most important issues. We therefore only use response sequences (i.e., a series of 0's and 1's showing the correctness of application of a particular skill performed by a particular student) as an input to the detector in the current study.

A higher level research question is if we can detect wheel spinning at all: Can we detect wheel-spinning only from the sequence of response accuracy? If so, how accurate the detection is? We hypothesize that if teachers can systematically identify the moment of wheel-spinning only by observing the correctness of student's response, then a neural-network model should be able to learn to detect the moment of wheel-spinning in the same way as teachers do.

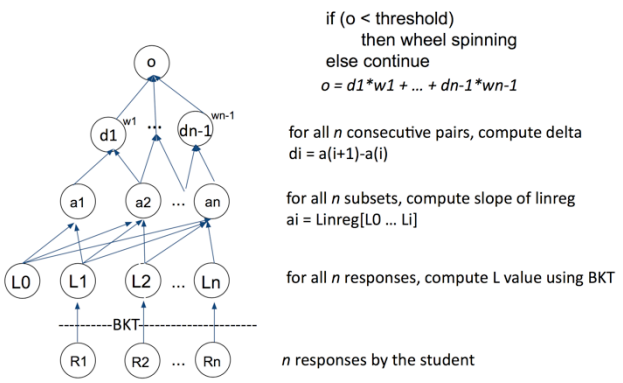
2. THE DETECTOR
Our basis for identifying wheel spinning is to analyze the correctness of student responses for a particular skill. We then attempted to test our hypothesis by comparing the predictions of our detector with examples classified by human coders. We asked two human coders to qualify the student's response data to identify wheel-spinning cases based on our coding manual. Table 1 shows a contingency table showing the agreement between two coders. The inter-coder reliability (the Cohen's kappa) on this final coding is 0.90.

Table 1. Inter-coder agreement of wheel-spinning coding

		Coder 2		
		W	C	Total
Coder 1	W	72	13	85
	C	5	752	757
	Total	77	765	842

Having identified the wheel spinning cases, we attempted to train a neural network to learn a latent pattern in a gradual change in a

sequence of 1s and 0s, representing the first attempt a student has a step for a certain skill.



The input of the NN-based detector is a *response sequence* (denoted as R_1, R_2, …, R_n in the figure) that shows a chronological record of the correctness of skill application made by the student on a particular skill. Each time a new response is observed (i.e., R_n in the figure), the response sequence is fed into the Bayesian Knowledge Tracer (BKT) to update a predicted mastery level up to the point of the latest response observation (denoted as L_1, L_2, …, L_n).

The first part of our neural network computes the change in the predicted mastery level represented as a slope of a linear regression model with the L value as a dependent variable and the opportunity count (i.e., i in L_i) as an independent variable. The slope of this line represents how gradual the student's learning has been. The second part of the neural network computes the deltas for each of the consecutive slope values. Students who are consistently learning have deltas greater than or equal to 0, because overall the trials that those students make forward progress. However, in the case of wheel spinning, the slopes decrease more often than they increase.

The output from the neural network is a weighted sum of the delta values (in the second hidden layer) representing the likelihood of wheel spinning. We train the neural network to learn weights for each delta values in such a way that the output less than zero indicates a potential of wheel spinning and the smaller the output value the more likely the student would wheel spin. The neural network updates weights using back propagation to converge on a set of weights that minimize the classification error during the training.

3. RESULTS

We used the dataset "Cog Model Discovery Experiment Spring 2010" in the study called "Geometry Cognitive Model Discovery Closing-the-Loop", taken from DataShop[1]. This dataset contained 5385 student-skill responses. Among 5385 student-skill response sequences, there are 2883 response sequences that have more than and equal to 5 responses. We filtered out response sequences with less than 5, because there would not be enough attempts to determine wheel spinning. Out of 2883, there are 842 response sequences that do not reach to the mastery according to BKT (hence potentially wheel spinning). In these 842 response sequences, there are 122 unique students and 44 unique skills included.

For our validation study, we decided to use only student-skill response sequences that had greater than or equal to 10 opportunities, because we were trying to find out the best number of opportunities to predict from 5 to 10. After filtering out instances with less than 10 attempts, we were left with 141 student-skill response sequences. We then randomly dropped one response sequence to have 140 student-skill response sequences for a 10-fold cross-validation. On the 9 folds training data, each of the skill-specific neural networks was trained until it classified training instances with the minimum classification errors. The accuracy of the prediction was computed as an overall average across 10 cross-validations. We computed a precision and recall score for each 10-fold-validation, along with a corresponding F1 score. Figure 3 shows precision, recall, and F1 (which is 2*P*R/(P+R) where P and R shows precision and recall respectively) scores for N = 5 to 10.

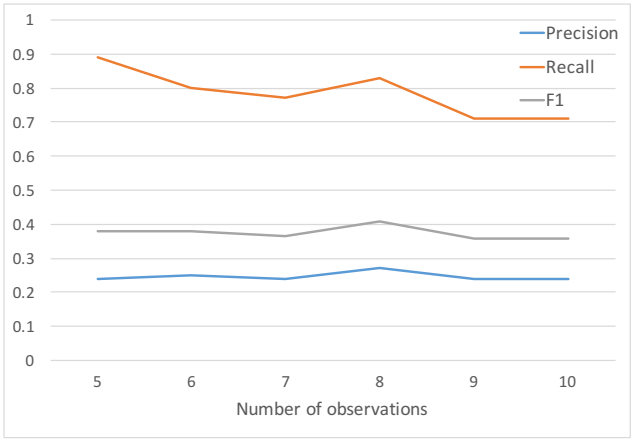


Figure 1. The precision, recall, and F1 scores computed on the first N response observations.

ACKNOWLEDGMENTS

The research reported in this paper has been supported by National Science Foundation Award No. 1418244.

REFERENCES

[1] BECK, J.E. and GONG, Y., 2013. Wheel-Spinning: Students Who Fail to Master a Skill. In *Artificial Intelligence in Education*, H.C. LANE, K. YACEF, J. MOSTOW and P. PAVLIK Eds. Springer Berlin Heidelberg, 431-440. DOI= http://dx.doi.org/10.1007/978-3-642-39112-5_44.

[2] CORBETT, A.T. and ANDERSON, J.R., 1995. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User Adapted Interaction 4*, 4, 253-278.

[3] RITTER, S., ANDERSON, J.R., KOEDINGER, K.R., and CORBETT, A., 2007. Cognitive tutor: Applied research in mathematics education. *Psychonomic bulletin & review 14*, 2, 249-255.

[4] VANLEHN, K., 2006. The Behavior of Tutoring Systems. *International Journal of Artificial Intelligence in Education 16*.

[1] https://pslcdatashop.web.cmu.edu

Exploring and Following Students' Strategies When Completing Their Weekly Tasks

Jessica McBroom, Bryn Jeffries, Irena Koprinska and Kalina Yacef
School of Information Technologies, University of Sydney, Sydney, NSW 2006, Australia
jmcb6755@uni.sydney.edu.au, {bryn.jeffries irena.koprinska kalina.yacef}@sydney.edu.au

ABSTRACT
In this paper, we explore methods of analysing data obtained from an autograding system involving weekly tasks and a finite set of possible strategies for completing these tasks. We present an approach to handling partially missing information and also investigate the usefulness of a sliding window rule mining technique in following changes in student strategy over time.

Keywords
Mining student behaviour and strategies, autograding system

1. INTRODUCTION
Teaching activities are often not offered in a linear way: it is sometimes useful to provide students with several choices of task, or to provide a gradual approach to learning by allowing a choice of tasks of varying difficulty. Maximum points could be achieved through implementing all the hard tasks, but students unsure of their ability might choose to take a more gradual approach, starting with the easy task and working up. We wish to understand how students manage their learning when presented with such choices by analysing the order in which students attempt such tasks. We investigate the following research questions: What strategies do students take in attempting the different tasks each week? Are there differences between the strategies of the regular and advanced students? In this work we report on several techniques applied to the data collected through an autograding system in a university database course. Our main contribution is in showing how to represent and mine data from student attempts of tasks with different levels of difficulty.

2. DATA
The data comes from weekly programming tasks in a third-year database course with students in a regular stream [2] (n=92), and an advanced stream [3] (n=20). Part of the assessment, for 10% of the final grade, was a set of weekly programming tasks for which students were required to implement various algorithms in Java and submit these implementations using the PASTA online submission platform 0. Tasks included skeleton code and unit tests, and students were encouraged to write and test their implementations locally before submitting. Once submitted to PASTA, the unit tests were applied again, and students received automated feedback of the outcomes of these tests. Students then had the option of submitting a revised attempt, or trying another of the three tasks, until the submission deadline had been reached. Each week there was a choice of three tasks with different levels of difficulty - easy, medium and hard. More marks were allocated for the more difficult tasks: 4 points for hard, 3 for medium, and 2 for easy tasks. Partial implementation of any task received 1 point. The data extracted from PASTA consisted of the marks for every student's attempt on each task.

3. STRATEGIES
There are 16 possible strategies that can be taken by a student for each weekly set of tasks: the 15 possible permutations of Easy (E), Medium (M) and Hard (H) tasks attempted, and no attempt at any task (None). Figure 1 shows the relative frequency of the different strategies taken by all students each week, and in total across all weeks. We labelled each strategy according to the order in which the tasks were completed. So, for instance, in the strategy EH a student completes that week's Easy task first, followed by the Hard task. Note though that this information is imperfect: students were only awarded marks for the most difficult task completed and had access to the unit tests at home, so may have completed multiple tasks while only submitting the most difficult of these. In addition, due to dependencies in tasks in some weeks, certain completion orders were forced. For example, in some weeks the medium task extended the easy task, so students were required to complete easy before medium. However, the most common strategies according to our data are None (30%), E (31%), EM (11%), EMH (15%), EH (4%), H (6%). The remaining 4% is a mixture of the other combinations with support less than 1%, including some where easier questions attempted later: we saw at least one instance of EHM, ME, MH, HE, HM and HME. Two strategies, MHE and HEM, were not observed at all.

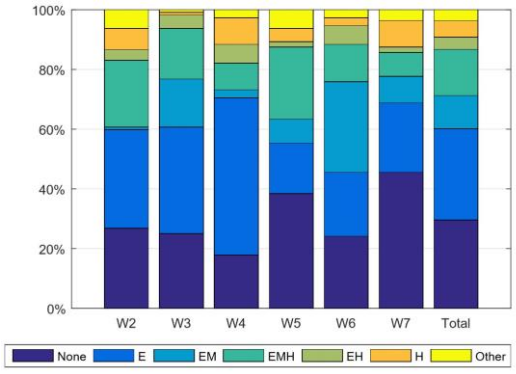


Figure 1. Relative frequency of strategies used by students in each week, and in total across all weeks

4. CLUSTERING
Since students had been allowed to test their code at home, we did not have access to perfect information about the order in which they completed the tasks. We therefore clustered students based

only on the highest difficulty task completed each week, ranking difficulties from 1 (easy) to 3 (hard). E.g., <3, 2, 3, … > would represent a student who completed the hard task in Week 2, the medium task in Week 3 and the hard task in Week 4. Using this representation we applied the *k*-means algorithm with *k*=5 (determined empirically). Cluster centroids are shown in Figure 2.

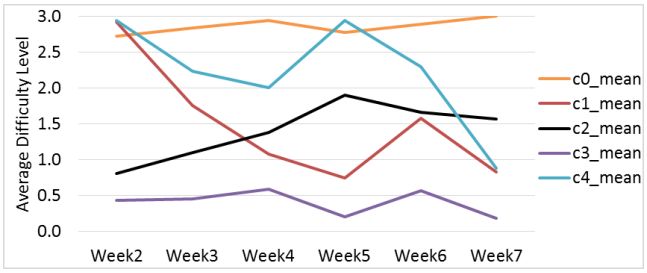


Figure 2. Average highest difficulty of tasks completed by students in each cluster, each week

We complemented the cluster analysis with the information on completion order which involved student submissions with and without potentially missing information. For example, if a student's strategy was EMH, then they definitely completed all three tasks. However, if a student's strategy was H, then they may have only completed the hard task, or they may have completed all three at home and only submitted the hard task. Since strategies with missing information were less frequent, we took the mode weekly strategy for each cluster as shown in 1, which allowed us to still compare student strategies despite the missing information.

Table 1. Mode weekly strategy per cluster. Last column shows proportion of regular and advanced (in parentheses) students.

Cluster	W2	W3	W4	W5	W6	W7	%(adv)
0	EMH	EMH	EMH	EMH	EMH	EMH	9(50)
1	EMH	EM	E	None	EM	None	13(0)
2	E	E	E	E	E	None	18(20)
3	None	None	E	None	None	None	45(15)
4	EMH	EM	E	EMH	EM	E	15(15)

We note that in some weeks there may have been dependencies between tasks that are ignored in this analysis. This limitation notwithstanding, we can broadly summarise behaviour in each clusters. Cluster 0 students complete the hardest task every week, by starting from the easy task and gradually progressing to the hardest task (EMH strategy). Cluster 1 students start well in Week 1 but then gradual drop in the difficulty of the completed tasks towards Week 7. Cluster 2 students start poorly but improve gradually, completing mainly easy tasks. Cluster 3 students consistently make very few submissions, and only of the lowest difficulty. Cluster 4 students generally perform well, often working through tasks of increasing difficulty but not always completing the medium or hard tasks. We speculate that Cluster 3 students may be investing little effort due to the relatively low weighting of the weekly tasks, while Cluster 4 students may have run out of time or found the later tasks too difficult to complete.

5. SLIDING WINDOW RULE MINING

To find trends in changes of strategy we looked for association rules X → Y in which X occurred before Y in time, since only these rules are likely to be of use. We further restricted our analysis to periods of three week. We extracted length-3 itemsets by using a sliding 3-week window over each student's strategy vector. Hence a student's 6-week behaviour vector <2EMH 3EM 4E 5EMH 6EM 7E> would generate 4 item sets <1EMH 2EM 3E>, <1EM 2E 3EMH>, <1E 2EMH 3EM>, <1EMH 2EM 3E>. This process is similar to rule mining in time-series subsequences [1], but here we encode the time into each item to allow us to use traditional association rule techniques.

Table 2. Highest-confidence rules found using length-3 sliding window rule mining technique

Rule	Support	Confidence	Lift
1None,2None → 3None	14%	85%	2.70
1EMH,2EMH → 3EMH	5%	62%	4.63
1EMH,2EM → 3E	3%	57%	2.00
1None,2E → 3E	3%	45%	1.58
1None,2E → 3None	3%	45%	1.43

From these item sets (n = 448) we searched for rules 1*a*,2*b* → 3*c* where *a*, *b* and *c* were the strategies used in consecutive weeks. The 5 highest confidence rules are shown in Table 2. The first rule shows that the likelihood of not attempting a task was very high if the student had not submitted two previous tasks. The second two rules suggest a student is likely to work through all three tasks progressively if they did so in the previous two tasks. Most other rules indicate that many students' strategies were on the borderline between completing the task only or none at all. Our technique was limited by task dependencies; we believe its effectiveness could be improved if applied to data without these deficiencies.

6. CONCLUSION

We have demonstrated how clustering can be applied to data from tasks in which students have choices between several activities, with a particular focus on handling missing information. We have also demonstrated how rule mining can elucidate trends in behaviour over a window of time, though the application of this technique was limited by missing information. These techniques were both limited by variability in dependencies in the different tasks, but still demonstrate how useful knowledge can be extracted from such data.

7. ACKNOWLEDGMENTS

This work was funded by the Human-Centred Technology Cluster of the University of Sydney.

8. REFERENCES

[1] Shokoohi-Yekta, M., Chen, Y., Campana, B., Hu, B., Zakaria, J., Keogh, E., 2015, Discovery of Meaningful Rules in Time Series. In *Proceedings of the 21st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 1085-1094

[2] INFO3404: Database Systems 2 (2015 - Semester 2). https://cusp.sydney.edu.au/students/view-unit-page/uos_id/125755/vid/309912. Accessed: 2016-05-20.

[3] INFO3504: Database Systems 2 (Adv) (2015 - Semester 2). https://cusp.sydney.edu.au/students/view-unit-page/uos_id/125790/vid/309914. Accessed: 2016-05-20.

[4] Radu, A. and Stretton, J. PASTA, School of Information Technologies, University of Sydney, http://www.it.usyd.edu.au/~bjef8061/pasta/. Accessed: 2016-05-20

Identifying Student Behaviors Early in the Term for Improving Online Course Performance

Makoto Mori
Department of Computer Sciences
Florida Institute of Technology, USA
mmori2013@my.fit.edu

Philip K. Chan
Department of Computer Sciences
Florida Institute of Technology, USA
pkc@cs.fit.edu

ABSTRACT
To study the correlation between student behavior and performance, we propose using high-level behavior features and a random forest algorithm. Considering a course with 10 periods, our results indicate that our models can reach 70% accuracy in the first period and 90% in the first 5 periods and starting to study earlier is important in individual behaviors and behavior combinations.

1. INTRODUCTION
The main goal of this study is to identify student behaviors in the first half of the semester that are correlated to strong performance so that we can provide feedback and encourage more appropriate behavior. The contributions of our study include: (1) we introduce *high-level* behavioral features derived from the course syllabus and sequential patterns; (2) we propose a random forest algorithm with cross-validation; (3) considering a course with ten periods, our empirical results indicate that our models can reach at least 70% accuracy from behavior features in the first cumulative period and 90% from features in the fifth cumulative period; (4) our approach can identify both important single behavior and behavior combinations. Our empirical results indicate that starting to access course materials early (a *high-level* feature) is important in individual behaviors and behavior combinations.

2. RELATED WORK
Many studies, e.g. [5], generally use how frequent activities occur and how long activities take as main features in their models. We call such features low-level features. Besides low-level features, related studies [4, 6] propose sequence of activities as features that come from a sequential pattern mining algorithm [4]. Further, Jo et al. [2] measure the interval of login sessions to find the regularity of login interval. Coffrin et al. [2] analyze the ordering of materials used in a course. We call features that not only simply measuring frequency and duration of activities as *high-level* features. For learning algorithms, many related studies, e.g. [8], use a single learning algorithm to predict student performance. However, Elbadrawy and Studham [3] propose using linear multi-regression, which is a weighted sum of multiple linear regression models. Many related studies perform performance prediction based on analysis using student activities from the entire term, which does not allow intervention during the term. Some related studies, e.g. [3], use non-behavior features such as quiz or assignment scores in their model. A number of studies only analyze individual behaviors separately. However, some studies analyze behavior combinations. Elbadrawy and Studham [3] use a weighted sum of multiple linear regression models, each of which can be considered as a behavior combination. Kinnebrew and Biswas [6] use SPAM [4] to identify important sequence of learning behaviors. Our approach uses high and low-level behavior features early in the term with an ensemble learning algorithm to identify both important single behaviors and behavior combinations.

3. APPROACH
In this study we focus on three steps. The first step is to generate features that can represent students' behavior. The second step is to use a machine learning algorithm to find correlations between behavioral features and performance. The third step is to identify important behaviors from the learned models.

3.1 Generating Features
Based on our experience, we identify *low-level* features that characterize the amount of different activities. Activities include number of logins, number of videos watched, number of questions asked and so on. ASRs (Active Student Responding Exercises) are questions that are embedded in the instructional video and students enter their answers after watching the video.

For *high-level* features, we focus on measuring beyond just "how frequent" or "how much" from the log files. For example, a motivated student would likely schedule a regular study time. To measure how regular a student studies, we first identify the day of the week that the student studies the most. For example, if a student studies most on Wednesdays, the student is quite regular in using Wednesday for studying. We then divide the frequency of the most studied weekday (e.g. Wednesday) by the frequency of the weekday (e.g. Wednesday) in the behavior period. The course syllabus has due dates and test dates. We generate features of student behavior with respect to those dates. For example, number of days the student studies before a test, number of days to submit a test before it is due. The syllabus also specifies when materials are released. We generate features that measure how soon the student starts accessing the released materials. We use SPAM [4] to identify high-level features based on behavior sequences. SPAM finds sequential patterns that meet the minimum support and maximum gap constraints. Support is the count of a sequence, while gap is the number of "wide cards" between items in a sequence.

3.2 Random Forests with Cross Validation
To improve effectiveness, we propose using the random forest algorithm [16] which builds multiple less-correlated decision trees and combines the classifications from individual trees. The random forest algorithm has two key parameters: forest size (number of trees) and feature subset size (number of features that can be considered in each node). To find a suitable combination of forest size and feature subset size, we vary the two parameters, build a forest, estimate the quality of the forest via cross validation (by splitting the training set), and select the parameter combination that yields the most accurate forest.

3.3 Identifying Important Behaviors
Given a random forest, we identify the most frequent feature used in the root nodes as the most important single behavior. In a random forest, the root of each tree is selected from a random subset of all the features. Hence, the most frequent feature in the root nodes is most likely to be the most important behavior.

Considering a single behavior might not be sufficient, we desire to study behavior combinations that are correlated with higher performance. Consider a forest that has n trees, we calculate a quality score for each feature combination that appears in the top two levels of a tree. The score of feature combination f_i in tree r is the number of positive examples $P_r(f_i)$ divided by the total number of examples $T_r(f_i)$ for this combination. The score of a feature combination $S(f_i)$ in the forest is the sum of scores from the trees: $S(f_i) = \sum_{r=1}^{n} \frac{P_r(f_i)}{T_r(f_i)}$.

4. EXPERIMENTAL EVALUATION

Our main task is to find important behaviors in the first half of the term that correlate with an above average score on the final exam. Also, we identify behaviors that we can encourage later, instead of just asking students to perform better on assignments and tests. Within the first half of the term, we would like to study how early we can identify important behaviors that estimate performance accurately. We divide the first half of the term into multiple periods (e.g. weeks). Features are generated from behavior in period 1 through *k*. We call such periods as "cumulative" periods.

This study analyzes BEHP5000 "Concepts and Principles of Behavior Analysis" that was offered in 2013 at Florida Institute of Technology. We obtained data for 110 students from the course. Our evaluation criterion is prediction accuracy on the test set. Two thirds of students are randomly selected to form the training set and the rest of students are in the test set. To generate sequential patterns with the SPAM algorithm, we use 70% as the minimum support and 2 as the maximum gap.

To compare the effectiveness of our proposed approach with existing approaches, we select a decision tree learning algorithm without and with rule post-pruning [7]. We also choose the original random forest algorithm [1] that uses 100 as the forest size, and log2**M* as the feature subset size, where *M* is the number of features. We use k=5 in the k-fold cross-validation for our random forest algorithm. For each k-fold cross-validation, we vary the forest size from 99 to 999 and the feature subset size from log2**M* to 55.

4.1 Predicting Performance on Final Exam

According to Figure 1, random forest with k-fold cross-validation is the most accurate among the four algorithms. Random forest based models are more accurate than other algorithms. Our approach reaches 74% of accuracy in the first cumulative period, and 90% of accuracy in the fifth cumulative period.

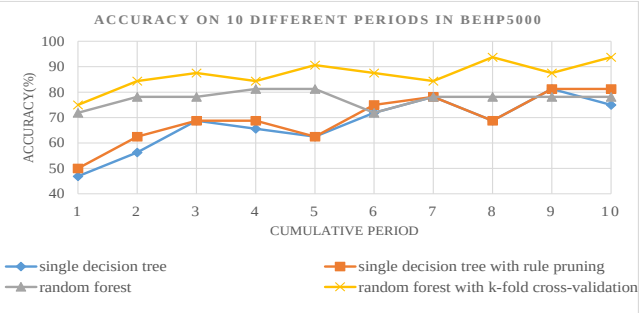


Fig. 1. Accuracy of 4 algorithms from 10 cumulative periods.

4.2 Important Student Behaviors

In the first half of the semester the most frequent feature is *days_after_unit_release* and appears in every cumulative period. This behavior measures, after the unit materials have been released, how many days the student takes to start accessing the materials. The behavior indicates how early a student starts to study, and hence, how motivated the student is. The second most frequent feature is *total(asr_times)* which appears 3 times. This behavior measures the number of times a student attempts ASR, which tries to improve student engagement and understanding of concepts presented in videos. More ASR attempts indicate a student is more engaged and yields deeper understanding.

The most frequent behavior combination is *total(days_after_unit_release)*>x and *test_submit_before_due* <=y which is marked in blue. Both features are high-level features. *total(days_after_unit_release)* represents how early the student starts to access to the unit material after it has been released. *test_submit_before_due* represents how early students submit test before the due date that is stated in the syllabus. Both features are highly related to study motivation of students. Smaller x and larger y values indicate higher motivation. That is, we expect *total(days_after_unit_release)* "<" x and *test_submit_before_due* ">=" y would indicate a highly motivated student. However, we found *total(days_after_unit_release)* ">" x and *test_submit_before_due* "<=" y is the most frequent. In other words, the student begins accessing the materials later and submits the test later, which is counter intuitive. One possible reason is that the behavior combination identifies a small group of students who are smart, therefore, they start studying later and submit test later. Another reason is that the behavior combination appears in cumulative periods 2 and 3, which include less data for the student behavior, therefore, the behavior combination might be less reliable.

Due to space limitation, further details can be found at: `cs.fit.edu/~pkc/papers/edm16long.pdf`.

5. REFERENCES

[1] Breiman, L., 2001. Random forests. *Machine Learning*, 45(1), pp.5-32.

[2] Coffrin, C., Corrin, L., de Barba, P. & Kennedy, G., 2014. Visualizing patterns of student engagement and performance in MOOCs. In *Proc. Intl. Conf. on Learning Analytics & Knowledge* (pp. 83-92).

[3] Elbadrawy, A., Studham, R.S. and Karypis, G., 2015. Collaborative multi-regression models for predicting students' performance in course activities. In *Proc. Intl. Conf. on Learning Analytics & Knowledge* (pp. 103-107).

[4] Ho, J., Lukov, L., & Chawla, S. 2005. Sequential pattern mining with constraints on large protein databases. In *Proc. Intl. Conf. on Management of Data (COMAD)* (pp. 89-100).

[5] Jo, I., Kim, D. & Yoon, M., 2014. Analyzing the log patterns of adult learners in LMS using learning analytics. In *Proc. Intl. Conf. Learning Analytics & Knowledge* (pp. 183-187).

[6] Kinnebrew, J. & Biswas, G., 2012. Identifying Learning Behaviors by Contextualizing Differential Sequence Mining with Action Features and Performance Evolution. Proc. *Intl. Conf. Educational Data Mining* (pp. 57-64).

[7] Mitchell, T., 1997. *Machine learning*. McGraw-Hill.

[8] Seaton, D.T., Bergner, Y., Chuang, I., Mitros, P. and Pritchard, D.E., 2014. Who does what in a massive open online course? *Comm. of the ACM*, 57(4), pp.58-65.

Time Series Analysis of VLE Activity Data

Ewa Młynarska
Insight Centre, University College Dublin, Ireland
ewa.mlynarska@insight-centre.org

Derek Greene
Insight Centre, University College Dublin, Ireland
derek.greene@ucd.ie

Pádraig Cunningham
Insight Centre, University College Dublin, Ireland
padraig.cunningham@ucd.ie

ABSTRACT

Virtual Learning Environments (VLE), such as Moodle, are purpose-built platforms in which teachers and students interact to exchange, review, and submit learning material and information. In this paper, we examine a complex VLE dataset from a large Irish university in an attempt to characterize student behavior with respect to deadlines and grades. We demonstrate that, by clustering activity profiles represented as time series using Dynamic Time Warping, we can uncover meaningful clusters of students exhibiting similar behaviors even in a sparsely-populated system. We use these clusters to identify distinct activity patterns among students, such as Procrastinators, Strugglers, and Experts. These patterns can provide us with an insight into the behavior of students, and ultimately help institutions to exploit deployed learning platforms so as to better structure their courses.

Keywords

Learning analytics, Data mining, Moodle, Time series, VLE

1. INTRODUCTION

The availability of log data from virtual learning environments (VLEs) such as Moodle presents an opportunity to improve learning outcomes and address challenges in the third level sector. We propose representing a student's efforts as a complete time-series of activity counts. We analyse yearly anonymised Moodle activity data from 13 Computer Science courses at University College Dublin (UCD), Ireland, and seek to identify patterns and relationships between more than one attribute that might lead to a student failing a course. A major potential benefit of this would be to introduce mechanisms identifying issues in the learning system early during the semester, supporting interventions and changes in the way in which a course is delivered.

A large amount of previous research in this area relates to different activity types, which are most predictive for a single dataset [1, 3]. This makes it difficult to generalise those methods to systems where the type and volume of Moodle activity can vary significantly. In order to facilitate the performance prediction on less structured systems, we need methods incorporating multiple features to deal with the sparsity problem. As a solution, we present a method for mining student activity on sparse data via Time Series Clustering. We explore the use of Dynamic Time Warping (DTW) as an appropriate distance measure to cluster students based on their activity patterns, so as to achieve clustering indicating more structured activity patterns influencing students' grades. DTW allows two time series that are similar but out of phase to be aligned to one another. To gain a macro-level view regarding whether these patterns occur across all assignments, we subsequently perform a second level aggregate clustering on the clusters coming from each assignment. This results in seven prototypical behaviour patterns (see example in Figure 1), that we believe can lead to better understanding of the behaviour of larger groups of students in VLEs.

2. TIME SERIES ANALYSIS

To perform clustering, the Moodle activity data was transformed into a series of equispaced points in time. In our case, a time series is a three week timeline – from two weeks before a given assignment submission date until one week after the deadline. These timelines were divided into 12 hour buckets of activity counts. We applied k-means clustering using DTW as a distance measure to cluster the timelines for each assignment. For a given number of clusters k, the algorithm was repeated 10 times and the best clustering was selected (based on the fitness score explained below). Due to the fact that DTW is not a true metric, k-means is not guaranteed to converge, so we limited each run to a maximum of 50 iterations. To choose the size of the DTW time window, we ran k-means for *window sizes* $\in [0, 3]$. The results did not conclusively indicate that any single *window size* leads to a significant decrease in cluster grade variance, which is unsurprising. In cases where there are many time series exhibiting little activity, it will be difficult to differentiate between the series and so a larger window size will be more appropriate. Based on this rationale, we believe that *window size* selection should be run for each assignment separately when applying this type of analysis in practice. The fitness function helping in selection of the best clustering needs to take into consideration that two clusters of different sizes might have the same variance value; this issue can be solved by applying a penalty to smaller clusters. We also

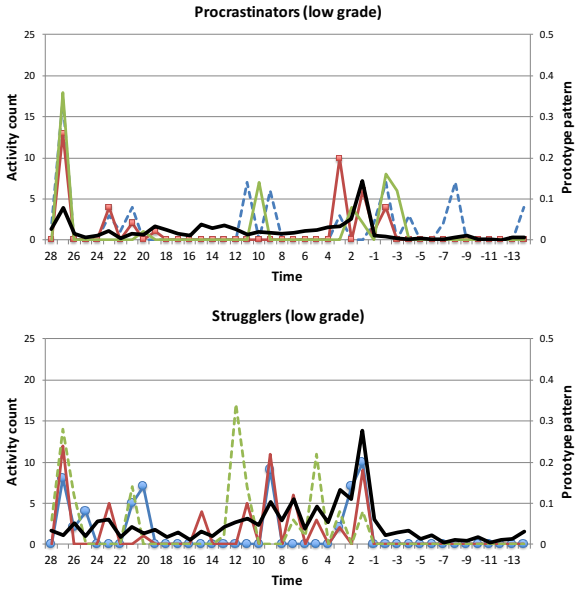


Figure 1: Two of the seven prototype activity patterns that occurred in Assignment #1. The black trend-line represents the prototype pattern. The coloured lines represent the activities of individual students. Negative numbers on the Time axis represent time after the deadline.

would like a "balanced clustering" where the variance of the cluster sizes is as small as possible. Based on these requirements, the fitness score calculation for a clustering generated by k-means consists of three steps:

1. The mean variance of the k-means clustering is calculated using the weighted average of all the clusters' variances, where the weight is based on the size of the cluster. This way the clusterings containing larger clusters with lower variances will be awarded better scores.
2. It is crucial to test the difference between a baseline clustering and actual results to define the significance of the clustering. For that purpose we run multiple random assignments of time series to calculate the expected score which could be achieved by chance for a given number of clusters.
3. To incorporate the baseline comparison in the score, the weighted average variance score from Step 1 is normalised with respect to the random assignment score from Step 2. A good clustering should achieve a low resulting score.

3. DISCUSSION

In our analysis, we took into account 52 two weeks assignments due to their longer and richer time series. We applied the time series clustering methodology described in previous section to the activity data for each of the assignments in the dataset, which are naturally split into two semesters. The Semester 1 clusterings appeared to show a number of frequently-appearing patterns across different courses. To gain a deeper insight into these patterns, we applied a second level of clustering – i.e. a clustering of the original clusters from all assignments. To support the comparison of clusters originating from different modules, the mean time series for each cluster was normalised. Based on the associated assignment scores, these normalised series were then stratified into low, medium, and high grade groups. We subsequently applied time series clustering with $k = 4$ and *window size* 1 to the normalised series in each of the stratified groups. Grade group names chosen by us were motivated by the behavioural pattern of students and some of them were inspired by previous research [2]. This second level of clustering revealed seven distinct prototypical patterns, which are present across multiple assignments and courses: *Procrastinators, Unmotivated, Strugglers, Systematic, Hard-workers, Strategists and Experts.*

The students rewarded with low grades were the second largest group of submissions after medium graded submissions having the smallest average activity per submission. The first out of 3 largest clusters was a group barely active on Moodle, performing submission activity at the deadline only (See Figure 1). As mentioned by Cerezo *et al.* [2], these could be labelled as Procrastinators. The black trend-line on the graph depicts prototype activity pattern and group of time series represents activity of students from the sample cluster. The third biggest group contains those students doing the minimum amount of work and showing larger activity towards the deadline (see Figure 1). The second academic semester courses mostly exhibit similar clusters from the first semester. The percentages indicate that for the Low Grade group, the Strugglers were most common and Procrastinators were less common.

While we did observe significant numbers of outliers, the relevant courses should be considered using a separate analysis to determine whether external factors are at play (e.g. continuous assessment rather than discrete assignments, lack of material provided on Moodle for a specific course). Finally, it is worth exploring anomalous clusters in the context of activity outside that assignment or course. We are currently in the process of extending our research to address the behavioural patterns of knowledge seekers in alternative, more complex learning environments.

Acknowledgments

This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289.

4. REFERENCES

[1] C. Brooks, C. Thompson, and S. Teasley. A time series interaction analysis method for building predictive models of learners using log data. In *Proc. 5th International Conference on Learning Analytics And Knowledge.*, ACM, 2015.

[2] R. Cerezo, M. Sanchez-Santillan, J.C. Nunez, and M.P. Paule. Different patterns of students' interaction with moodle and their relationship with achievement. In *Proc. 8th International Conference on Educational Data Mining*, 2015.

[3] L. V. Morris, C. Finnegan, and S. Wu. Tracking student behavior, persistence, and achievement in online courses. *The Internet and Higher Education*, 8.3:221–231, 2005.

Massively Scalable EDM with Spark

Tristan Nixon
Institute for Intelligent Systems
University of Memphis
365 Innovation Drive
Memphis, TN, USA, 38152
t.nixon@memphis.edu

1. INTRODUCTION

The creation and availability of ever-larger datasets is motivating the development of new distributed technologies to store and process data across clusters of servers. Apache Spark has emerged as the new standard platform for developing highly scalable cluster computing applications. It offers a wide range of connectors to numerous databases and enterprise data management systems, an ever growing library of machine-learning algorithms and the ability to process streaming data in near-realtime. Developers can write their applications in Java, Scala, Python and R. Applications can be run locally (for easy development and testing), and deployed to dedicated clusters or on clusters leased from cloud-computing providers.

2. TUTORIAL

This day-long tutorial will provide a hands-on introduction to developing massively scalable machine learning and data mining applications with Spark. Participants will be expected to follow along with all examples on their own laptops throughout the tutorial, and to collaborate in small groups. All code used in the tutorial will either be taken from publicly available examples, or be available for download from the IEDMS github repository[1], and made available under a very liberal open source license. All examples will be designed to process a modestly sized sample of the KDD cup dataset available from the PSLC DataShop[2].

In advance of the day, participants will be given instructions on how to install and configure Spark and Scala on their laptops, so that they might arrive at the tutorial ready to begin. Throughout the tutorial, participants will be given exercises and problems to solve in small groups. This will give them experience with the material as it is presented and hands-on practice with structuring a distributed application in Spark.

2.1 Outline

The following material will be covered in the course of the tutorial:

- An overview and history of cluster computing and the development of map-reduce
- An example of a very simple map-reduce algorithm (distributed word-count) in Spark
- An introduction to the Spark runtime model, including:
    - Basic import and export operations
    - Resilient distributed datasets (RDDs)
    - RDD transformations and actions
    - How Spark optimizes the execution of distributed computation
- An overview to the different deployment options for Spark, including:
    - Launching and using the interactive spark command-line shell program
    - Running spark programs locally on a single machine
    - Launching a Spark cluster on Amazon Web Services
    - Submitting applications to remote clusters
- An introduction to Spark streaming
- An introduction to SparkSQL and working with DataFrames
    - How to load and manipulate an EDM dataset (KDD cup data)
    - Data representations needed to fit various EDM algorithms
- An introduction to Spark's Machine learning library MLib, including:
    - Transformers and Estimators
    - Chaining transformers into machine-learning pipelines
    - Examples of common EDM algorithms in Spark:
        - IRT algorithms using logistic regression (AFM, PFM, IFM)
        - BKT parameter fitting: (brute-force, HMMs)

Any remaining time will be devoted to discussing potential applications that participants may have in mind for their own data or projects.

[1] https://github.com/IEDMS/spark-tutorial

[2] https://pslcdatashop.web.cmu.edu/KDDCup/

Study on Automatic Scoring of Descriptive Type Tests using Text Similarity Calculations

Izuru Nogaito
KDDI R&D Laboratories Inc.
3-10-10 Iidabashi,
Chiyoda-ku, Tokyo, 102-8460 Japan
+81-3-6678-1609
iz-nogaito@kddilabs.jp

Keiji Yasuda
KDDI R&D Laboratories Inc.
3-10-10 Iidabashi,
Chiyoda-ku, Tokyo, 102-8460 Japan
+81-3-6678-1609
ke-yasuda@kddilabs.jp

Hiroaki Kimura
KDDI R&D Laboratories Inc.
3-10-10 Iidabashi,
Chiyoda-ku, Tokyo, 102-8460 Japan
+81-3-6678-1609
ha-kimura@kddilabs.jp

ABSTRACT

In this paper, we evaluate the automatic scoring of a descriptive type test. In the experiments, three test similarity measures are compared in terms of automatic scoring quality. Two of them are BLEU and RIBES, which are *n*-gram and word-level matching processes respectively, originally used for automatic evaluation of machine translation output. The other similarity process is Doc2Vec, which utilizes distributed representation to calculate the cosine distance. It was finally found that, according to the experimental results, the most efficient process used to calculate the text similarity depends on the type of the question.

Keywords

Doc2Vec, BLEU, RIBES, Text Similarity, auto-scoring

1. INTRODUCTION

Recently, the importance of "21st Century Skills" has been advocated in educational circles. A descriptive type of test is one of the methods to measure this skill; hence, this type of test is becoming more important than a multiple choice test.

In this paper, we carried out experiments on automatic scoring of a descriptive type test. There are two types of methods for automatic descriptive type test scoring. The first method is a similarity-based method, which computes the similarity between a student's answer and a model answer. The second method does not require a model answer; however, it requires several natural language processing (NLP) tools that compute cohesion, coherence, etc. [1]. In this research, we adopt the first approach because our target language for automatic scoring is Japanese and some of the NLP tools are not supported in Japanese. Furthermore, our research partner could provide test items and model answers. In this paper, section 2 describes similarity measures that are used for automatic scoring. Section 3 demonstrates the experiments and their corresponding results, and finally, section 4 describes the conclusions and future work.

2. SIMILARITY MEASURES

In this research, we apply two similarity measures based on surface expression. Both of them were proposed for automatic evaluation of machine translation output. We also apply the similarity measures in a distributed expression to the automatic scoring experiments. In this subsection, we explain these similarity measures.

2.1 Similarity in surface expression

BLEU [2] is proposed for the evaluation of machine translations. It uses *n*-gram matching between a reference sentence and a machine translation output. A sentence that is shorter compared to the reference is penalized in the BLEU score calculation.

RIBES [3] is also an automatic evaluation measure for machine translations. First, it compares the machine translation output with a reference at the word level. Then, it inspects the word order for common words based on the rank correlation coefficient.

2.2 Similarity in distributed expression

Recently, by using deep learning technology, a word or sentence can be converted into a distributed expression that is a vector of several hundred dimensions. According to previous research [4, 5], the cosine similarity between the distributed expressions is fairly close to a semantic similarity. In this research, the gensim [1] version of Doc2Vec is used to build the model that converts the document into a distributed expression.

Table 1: Statistics of the Training Corpus for Doc2Vec

	# of words	Lexicon size
Japanese wiki abstract (WIKI)	29,944,313	1,398,558
Mainichi-News-Paper (1991-2014) (NP)	504,844,192	5,578,327
WIKI + NP	534,788,505	6,376,935

3. EXPERIMENTS

3.1 Experimental settings

Doc2Vec requires a text corpus for model training. For the experiments, we use a Wikipedia corpus (WIKI) and a Mainichi Newspaper corpus (NP). In addition, three models are trained: one using WIKI, one using NP and one using both WIKI and NP. Then, the best model is chosen for each test item in terms of the automatic scoring performance. Table 1 demonstrates the statistics of each particular corpus. In the experiments, we use ten

[1] https://radimrehurek.com/gensim/

test items.

Table 2 Answer Text-Data Specification

Item ID	Topic of question	Question type	Ave. length of student answers (words)	Lexicon size of student answers	Number of students
ID01	Book	Graph reading	112.2	62.5	21
ID02	Fisherman	Summarization	49.7	33.4	21
ID03	Food	Graph reading	96.4	49.0	24
ID04	Fishery	Graph reading	87.8	53.5	22
ID05	Supermarket	Summarization	101.4	59.7	22
ID06	University	Summarization	110.7	71.6	20
ID07	Japanese	Summarization	77.7	46.8	32
ID08	Mail	Summarization	58.9	44.6	42
ID09	Vietnam	Graph reading	57.5	31.2	29
ID10	Beef	Graph reading	90.2	44.2	24
ID01-10	Average		84.3	49.6	25.7

All test items are answered by at least twenty students, aged between 10 and 16 years. Each question has its own target grade. Table 2 demonstrates the data set. In the table, "Graph reading" indicates the situation where the students are asked to describe a fact that can be read from the given graphs. Normally this type of question is a short sentence. Further, "Summarization" indicates the situation where the students are asked to summarize a given text between 300 to 800 words long. In each test item, four model answers are made by four teachers. Each answer is also scored by four teachers. Averaged scores are used as the recorded evaluation results in the experiments.

3.2 Experimental results and Discussion

Figure1 shows the correlation between the subjective score and automatic similarity. For Doc2Vec, we trained models with three conditions: Newspaper corpus only (D2V/NP), Wikipedia corpus only (D2V/WIKI) and both Newspaper and Wikipedia (D2V/NP + WIKI).

The methods that use similarity in surface expression are partly advantageous in the summarization question type. In this type of question, students tend to use the expression in the given question sentence, and the variety of their word choice is small. Thus, the possibility of matching words on the model answer could be high. In fact, the correlation values of BLEU and RIBES for ID02, ID05, ID06, ID07 and ID08 are relatively high.

The methods that use similarity in distributed expression are partly advantageous for the automatic scoring of graph reading questions. In general, the answer for this kind of question has a wide variation of words because students are free to choose their own words.

Both types of results, however, are shown on the graph of reading questions. First, the correlation value from Doc2Vec is better than the other methods for ID03, ID04 and ID10. This is due to the reason described previously. Second, the value of Doc2Vec is inferior for ID01, though it is a graph reading question. In this case, we understand that the corpus used does not share many similar words with the model answer sentences. The result also shows that the Doc2Vec similarity sometimes also works as a complemental similarity.

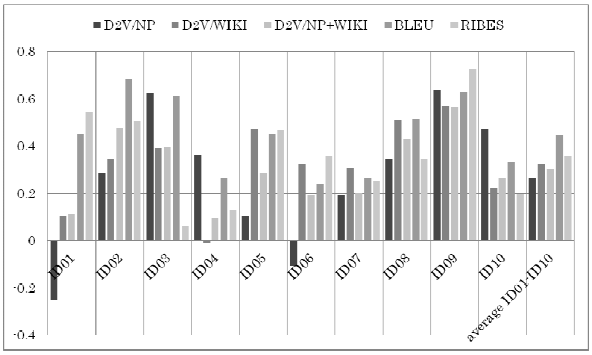


Figure 1 Correlation between subjective score and automatic method

4. CONCLUSIONS AND FUTURE WORK

For automatic scoring, we compared the Doc2Vec, the BLEU, and the RIBES similarities. In the case where the answers include a wide variation of words among students, the method using distributed expression seems to be more advantageous.

In future work, we will conduct research to use several similarities in a complementary way. We will also compare several methods, including the method using cohesion and coherence [1] that is described in the introduction section as a second method.

5. ACKNOWLEDGMENTS

This work uses model answers, student's answers, and scoring data that came from the Lojim clam school. (http://lojim.jp/).

6. REFERENCES

[1] Scott A. Crossley, Danielle S. McNamara.: Cohesion, coherence, and expert evaluations of writing proficiency, Proc. of the 32nd annual conference of the Cognitive Science Society, pp. 984-989, 2010.

[2] Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J.: BLEU: a method for automatic evaluation of machine translation, in Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL), pp. 311–318 (2002)

[3] Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, Hajime Tsukada.: Automatic Evaluation of Translation Quality for Distant Language Pairs, Conference on Empirical Methods on Natural Language Processing (EMNLP), Oct. 2010.

[4] Tomas Mikolov and Kai Chen and Greg Corrado and Jeffrey Dean.: Efficient Estimation of Word Representations in Vector Space, http://arxiv.org/pdf/1301.3781.pdf

[5] Quoc Le, Tomas Mikolov.: Distributed Representations of Sentences and Documents, http://arxiv.org/abs/1405.4053

Equity of Learning Opportunities in the Chicago City of Learning Program

David Quigley*, Ogheneovo Dibie, Arafat Sultan, Katie Van Horne, William R. Penuel, Tamara Sumner
University of Colorado Boulder
Boulder, CO 80309-0594
*david.quigley@colorado.edu

Ugochi Acholonu, Nichole Pinkard
Digital Youth Network
2320 N Kenmore Ave
Chicago, IL 60614

ABSTRACT

A novel method for understanding the equity of extracurricular learning opportunities within a regional learning ecosystem is presented. We apply the ecosystems concepts of abundance, richness, and evenness to understand the distribution of learning opportunities within the Chicago City of Learning. This analysis highlights the differences in learning opportunities across different neighborhoods the city. This article includes discussion of the ways these analyses can be used as a starting point for understanding city-wide informal learning communities.

1. INTRODUCTION

This work uses computational approaches to understand the spatial distribution of informal learning opportunities available to youth within the Chicago City of Learning (CCOL), a unique partnership and infrastructure built around supporting youth access to learning opportunities outside of school. Local organizations list their program offerings on the CCOL website and place them in one or more of eleven learning areas such as sports, science, or design. Youth access the site to browse and sign up for these programs. Our aim is to understand the degree to which these afterschool and summer opportunities are accessible to youth. The accessibility of programs relative to where youth live is a matter of *spatial equity* [4].

This research reports on the first year of efforts by CCOL members to document summer informal learning opportunities in Chicago, which resulted in over 4500 searchable learning opportunities. We developed a novel theoretical framework, inspired by concepts from the study of biological ecosystems, that draws on concepts of species richness, abundance, and evenness, and extends these concepts to characterize learning opportunities in a geographic space. We developed data mining approaches for operationalizing these concepts, drawing on data collected through the CCOL system. We present the theory, data mining approaches, and results on a specific question of interest: How are learning activities distributed across different neighborhoods in Chicago?

2. THEORETICAL FRAMEWORK

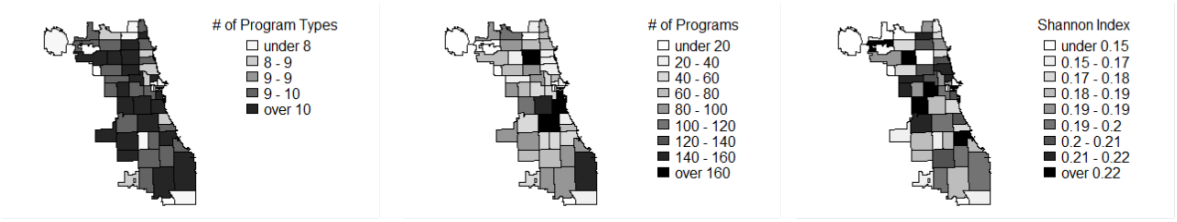
This framework extends Barron and colleagues' descriptions of learning ecologies as linked contexts that provide youth opportunities for learning (e.g. [1]). Human and ecological systems are constantly adapting to changing conditions, including conditions brought about by human activities. Resilient natural ecosystems - that is, ecosystems that have the capacity to adapt to a wide range of unexpected changes - are ones that have both an abundance of organisms and diversity of species [5]. Abundance refers to the number of organisms of a particular species in an ecosystem. Species diversity can be measured in two different ways: species richness and species evenness. Richness is a measure of the number of different kinds of organisms present in a particular area. Evenness measures the relative abundance of each species, or how close in numbers each species in an area are to the others.

These ideas about ecosystems have direct relevance to the study of youths' learning opportunities at the scale of a city. Young peoples' learning pathways are embedded within larger ecosystems of opportunity (e.g. [2]), and these concepts help describe those ecosystems. As in nature where all individual organisms are unique, each program is unique in the learning opportunities it provides to young people. Here, richness, abundance, and evenness refer to program offerings in different neighborhoods, where each individual program is analogous to an individual organism in an ecosystem, a program type is analogous to a species, and a neighborhood is considered a distinct ecosystem.

3. DATA SOURCES AND ANALYSIS

Our team analyzed programs offered through the CCOL website during the summer of 2014, from June 1st to September 30th. We extracted two pieces of information about each program: the program type and the program location. Program type refers to the eleven categories assigned within the CCOL system. Program location is the address of the program as entered by the hosting organization. We normalized the address of each program into a consistent format. We analyzed 3,931 face-to-face scheduled programs at 755 unique locations within the city limits of Chicago.

Program richness provides us with a way to characterize the diversity of opportunities, namely the degree to which program offerings of many different types are accessible from a particular neighborhood. This is determined for each zip code by counting the number of program types that have at least one program hosted in that area. Program abundance refers to the total number of unique programs within a given zip code. Program evenness allows us to measure the degree to which programs of a particular type may predominate



(a) Richness (b) Abundance (c) Evenness

Figure 1: Heatmaps of richness, abundance, and evenness metrics for zip codes in Chicago

in a neighborhood. This measure considers both the types of programs that are accessible and the overall number of programs of each type. We calculated evenness using the Shannon index, the same formula for species evenness in the study of ecosystems [3]. The Shannon index gives an evenness score from zero to one.

4. RESULTS

Figure 1a shows the richness metric - the number of program types with at least one program offering - for each zip code. Our analysis shows that many zipcodes exhibit high richness, with 39 out of the 59 zipcodes having programs spanning 9 or more of the 11 possible program types. Only one zipcode had a single program type being offered.

While many zipcodes exhibit richness, program abundance and program evenness tell a different story. Figure 1b shows the abundance - the total number of program offerings across all program types within each zip code. Here, we see that many of the programs are clustered in certain areas within the city. The large number of programs just south of downtown in particular highlights a hub of programs at cultural institutions such as museums. Other areas, such as the lakefront zip codes north of downtown, host fewer local programs on the CCOL site. Figure 1c shows the program evenness - demonstrated by Shannon index metrics - for each zip code. The indecies in all zip codes are relatively low (0 - .234), showing that all areas' offerings are skewed towards certain categories, rather than hosting a strong representation of programs of all types. In addition, program evenness has a degree of variance between zip codes in the city. Areas west of downtown show slightly better evenness scores than many of those to the south. This metric helps shed further light on the abundance figures shown in 1b. Though the area immediately south of downtown has high measures of abundance, the evenness scores in those same zip codes are lower than scores found in other parts of the city.

5. DISCUSSION

This work establishes a strong understanding of the distribution of learning programs across the city of Chicago. In some areas, cultural institutions are providing many programs in their area, which can skew the evenness metrics in those areas. In others, there are simply relatively few programs being offered. These results illustrate the utility of a data-driven ecological framework for analyzing the distribution of informal learning opportunities within a large urban environment. As the abundance, richness, and evenness heatmaps illustrate, no one metric is sufficient, as each captures different aspects of the larger ecosystem. These three measures, when visualized through the heatmaps in figure 1, provide a concise way to understand distribution of different learning opportunities across the city.

It is important to note the limitations of this approach. First, we used zip codes as our distinct ecosystem boundaries. Some zip codes cover large spaces and have odd shapes, so the presence of a program within that zip code is only a rough proxy of accessibility. Local transit infrastructure can have a significant impact on how well a learner can access a program, even if that program is hosted on the other side of the city. Also, this analysis covers only the first summer of operations of the the CCOL. As such, it is very likely that many learning opportunities taking place in churches, community centers, and other locales are not yet represented in the system. Thus, this analysis presents a single snapshot of only a portion of the total opportunities available to youth in the city.

6. ACKNOWLEDGMENTS

We would like to thank the Chicago Community Trust, the University of Colorado Boulder, and the entire CCOL team for supporting this research. This work would not have been possible without the generous technical support and data sharing provided by the CCOL team.

7. REFERENCES

[1] Barron, B., Gomez, K., Pinkard, N., and Martin, C. K. *The Digital Youth Network: Cultivating digital media citizenship in urban communities*. MIT Press, 2014.

[2] Holland, D., and Lave, J. Social practice theory and the historical production of persons. *Actio: An International Journal of Human Activity Theory*, 2 (2009), 1–15.

[3] Shannon, C. E., and Weaver, W. *The mathematical theory of communication*. University of Illinois press, 1998.

[4] Talen, E. School, community, and spatial equity: An empirical investigation of access to elementary schools in west virginia. *Annals of the Association of American Geographers 91*, 3 (2001), 465–486.

[5] Walker, B., and Salt, D. *Resilience thinking: sustaining ecosystems and people in a changing world*. Island Press, 2012.

Novel features for capturing cooccurrence behavior in dyadic collaborative problem solving tasks

Vikram Ramanarayanan
Educational Testing Service R&D
90 New Montgomery St, #1500
San Francisco, CA
vramanarayanan@ets.org

Saad Khan
Educational Testing Service R&D
600 Rosedale Road
Princeton, NJ
skhan002@ets.org

1. INTRODUCTION

Research shows that complex interactive activities such as team work and collaboration are more effective when participants are not only engaged in the task but also exhibit behaviors that facilitate interaction [5]. Successful collaboration is often manifested in what is known as "entrainment" or convergence between the participants of such collaboration. In the educational context, entrainment between collaborators or between student and the tutoring system is important in understanding learning dynamics, learning gains and student performance in different learning environments [6]. Recently Luna Bazaldua et al. demonstrated a statistically significant synchronicity of cognitive and non-cognitive behavior between dyads engaged in online collaborative activity [1]. However, in their study participants were not able to see each other and only interacted over a text-based chat interface. This is an important point to note since the ability to converse face-to-face can significantly impact the nature of the dyadic interaction. Therefore, in this paper we focus on behavioral patterns of emotional expressions between dyads during face-to-face conversation through a video conferencing system. Our hypothesis is that dyads engaged in face-to-face collaborative activity demonstrate a significantly different pattern of behavior as opposed to nominal dyads who are artificially paired up with each other. Notation-wise, we use the term nominal dyad or artificial dyad interchangeably to mean two subjects whose data are analyzed as if they were interacting dyadically, but were actually not.

Explicitly modeling temporal information in such dyadic interaction data is important because each person's emotional state or behavior need not stay constant over the course of the interaction – they could get fatigued over time, or be more nervous at the very beginning (resulting in repetitive, cyclic fidgeting behavior), but gradually settle into a comfort zone later, as they get more familiar with the task and each other. For similar reasons their body language and emotional state can also fluctuate over the time series. However, current feature extraction approaches that aggregate information across time do not explicitly model temporal cooccurrence patterns; consider for instance that one person's emotional state – joy – generally follows his interlocutor's emotional state – say neutral – in a definitive pattern during certain parts of the interaction. Capturing such patterns might help us (i) explicitly understand the predictive power of different features (such as the occurrence of a given pair of emotions) in temporal context (such as how often did the emotional state of one person in the dyad occur given the previous occurrence of another emotional state of the other person in the dyad), thus allowing us to (ii) obtain features that are more interpretable on visual inspection. We would like to take an initial stab at bridging this gap in this paper. Specifically, we propose to adapt a feature based on histograms of cooccurrences [4] that was developed earlier for analyzing a single time-series (say, from one person), and extend it to the case of dyads (see Figure 1). The feature models how different "template" emotional states of one person in a dyad co-occur within different time lags of a "template" emotional states of the other person in the dyad over time. Such a feature explicitly takes into account the temporal evolution of emotional states in different interaction contexts.

2. DATA

2.1 The Tetralogue CPS Platform

We used an online collaborative research environment developed in-house – the Tetralogue [2, 1]. The participants, who may be in different locations, interact through an online chat box and system help requests (selecting to view educational videos on the subject matter). The main avatar, Dr. Garcia, introduces information on volcanoes, facilitates the simulation, and requires the participants to answer a set of individual and group questions and tasks. A second avatar, Art, takes the role of another student who shows his own answers to the questions posed by Dr. Garcia, in order to contrast his information with that produced by the dyad. Twenty-six subjects participated in this study and were paired in dyads using random selection.

3. ANALYSES AND OBSERVATIONS

In order to observe how well HoC features capture dyadic behavior, we randomly extracted 100 time-intervals (each 10 seconds long) from the post-processed and synchronized feature streams for all 26 subjects. We then computed HoC features for each of these intervals for each subject, respectively. Now recall that in this pool of subjects, each subject has one true dyad with whom they completed the Tetralogue task collaboratively. We hypothesize that the HoC features computed for true dyads will be significantly different as compared to the HoC features computed between artificial or nominal dyads (who did not actually engage in a dyadic interaction). We found that the distances computed between HoC features extracted from true dyads were significantly lower ($p \approx 0$) than those of distances between HoC features computed on artificial dyads. This finding suggests that (i) not only do true dyads engaged in a

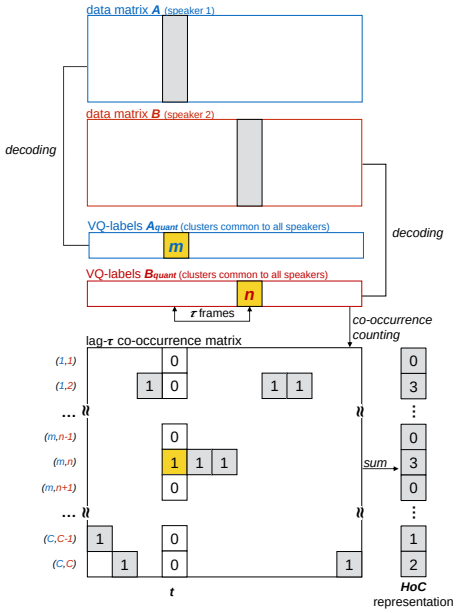


Figure 1: Schematic depiction of the computation of histograms of cooccurrences (HoC) (adapted from [3]).

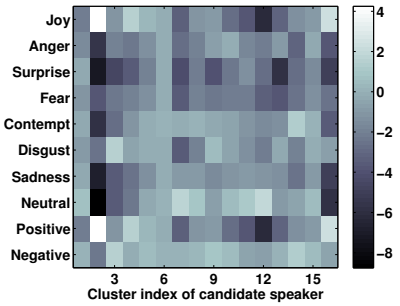


Figure 2: Schematic illustrations of the emotion feature clusters computed for *all* speakers. Each column represents an emotional cluster centroid, which is a particular distribution of emotional state activations. There are 10 dimensions that describe an emotional state, represented by different rows. The colors represent the odds, in logarithmic (base 10) scale, of a target expression being present (typically range: $[-5,+5]$).

collaborative interaction exhibit specific characteristic patterns of emotional state cooccurences that clearly sets them apart from artificial dyads, but (ii) such HoC features allow us to capture these differences in an effective manner.

Figures 2 and 3 gives us some more insight into why these features perform well. Figure 2 depicts the 16 cluster centroids computed on (and therefore common to) all speakers. Notice that each column of Figure 2 represents one cluster centroid, comprising different relative activation of different emotions – for instance, cluster 2 represents an emotional state with a higher activation of joy and positive emotion, while cluster 6 represents a more neutral emotional state, encompassing an equal (and approximately zero) activation of all emotions. Recall that these emotion clusters are common to *all* speakers. Figure 3 shows feature distributions of HoC features computed on one particular speaker and his/her actual dyadic partner, and those computed on that same speaker and an artificial dyadic partner. We observe that the feature distributions of the former are more peaky, with specific certain clusters of emotions

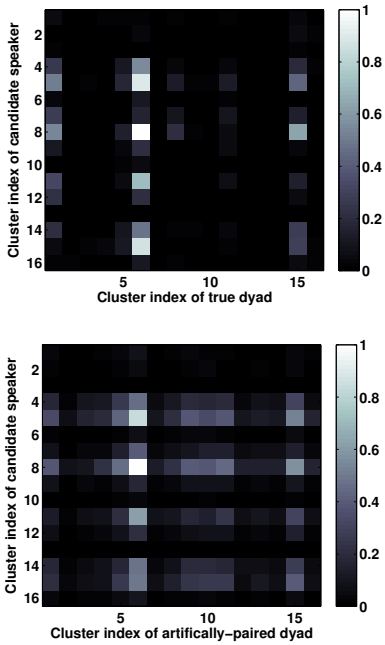


Figure 3: Average HoC feature distributions (across lags) for the true and nominal dyad, respectively, of one particular speaker in the database. The color in the $(m,n)^{th}$ square represents the average normalized activation (between 0 and 1) of cluster m of the speaker represented along the y-axis co-occuring with cluster n of the speaker represented along the x-axis.

co-occurring more often than others. However, in the case of the latter, this distribution is more flat and uniformly distributed. Note that while specific results shown in Figure 3 are particular to the chosen speaker, we observe the aforementioned trends are in general for all speakers. In other words, true dyads display specific patterns of behavioral cooccurrence and synchronicity that are not observed in artificial dyads, and such a HoC feature is helpful in understanding and bringing out these differences.

4. CONCLUSIONS AND OUTLOOK

This paper has made an initial attempt at proposing a novel feature, dubbed histograms of cooccurrences, that captures how often different prototypical behavioral states exhibited by one person co-occur with those exhibited by his/her partner over different temporal lags. We have shown that not only does this feature bring out the differences between dyads and non-dyads, but is also interpretable in that it tells us which behavioral states are most likely to occur in dyads as opposed to non-dyads.

5. REFERENCES

[1] D. L. Bazaldua, S. Khan, A. von Davier, J. Hao, L. Liu, and Z. Wang. On convergence of cognitive and non-cognitive behavior in collaborative activity. In *The 8th International Conference on Educational Data Mining (EDM 2015)*.

[2] L. Liu, J. Hao, A. A. von Davier, P. Kyllonen, and D. Zapata-Rivera. A tough nut to crack: Measuring collaborative problem solving. *Handbook of Research on Technology Tools for Real-World Skill Development*, page 344, 2015.

[3] V. Ramanarayanan, C. W. Leong, L. Chen, G. Feng, and D. Suendermann-Oeft. Evaluating speech, face, emotion and body movement time-series features for automated multimodal presentation scoring. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 23–30. ACM, 2015.

[4] V. Ramanarayanan, M. Van Segbroeck, and S. Narayanan. Directly data-derived articulatory gesture-like representations retain discriminatory information about phone categories. *Computer Speech and Language*, 36:330–346, 2016.

[5] A. A. Tawfik, L. Sanchez, and D. Saparova. The effects of case libraries in supporting collaborative problem-solving in an online learning environment. *Technology, Knowledge and Learning*, 19(3):337–358, 2014.

[6] J. Thomason, H. V. Nguyen, and D. Litman. Prosodic entraiment and Tutoring Dialogue Success. In H. Lane, K. Yacef, J. Mostow, and P. Pavlik, editors, *Artificial Intelligence in Education, AIED 2013*, pages 750–753. Springer, 2013.

Adding eye-tracking AOI data to models of representation skills does not improve prediction accuracy

Martina A. Rau
Department of Educational Psychology
University of Wisconsin—Madison
1025 W. Johnson St
Madison, WI 53706
+1-608-262-0833
marau@wisc.edu

Zach Pardos
2nd author's affiliation
1st line of address
2nd line of address
Telephone number, incl. country code
2nd E-mail

ABSTRACT
Visual representations are ubiquitous in STEM instruction. Representation skills allow students to use visual representations to learn about concepts. It seems reasonable to hypothesize that we can gather useful information about representation skills from eye-tracking AOI data that assesses how students pay attention to representations. We tested this hypothesis by comparing cognitive models with and without eye-tracking AOI data. Specifically, we used Bayesian Knowledge Tracing and Long Short Term Memory models. We evaluated these models based on their accuracy in predicting students learning of knowledge components that assess representation skills. Eye-tracking AOI data did not improve the prediction accuracy of our cognitive models. We compare our results to prior research to generate hypotheses for future research.

Keywords
Visual representations, intelligent tutoring system, eye-tracking, Bayesian Knowledge Tracing, Long Short Term Memory models.

1. INTRODUCTION
STEM instruction typically uses visual representations that depict to-be-learned content [1]. To learn content knowledge, students have acquire *representation skills*: the ability to use visual representations to learn [2]. Instructional support is most effective if it not only focuses on students' learning of content knowledge, but also on their learning of representation skills [1]. Intelligent tutoring systems (ITSs) have the capability to adapt to the individual student's needs [3]. They do so based on a cognitive model that infers the student's knowledge level based on interactions with the ITS [3]. Hence, the goal of cognitive modeling is to accurately model students' learning in real time [4]. A limitation of this research is that it has mostly focused on students' content knowledge, not on representation skills.

It seems reasonable to assume that we can gather useful information about students' learning of representation skills from their visual attention to representations [5]. However, most prior eye-tracking research involved relatively simple learning materials; typically expository text paired with one additional visual representation. By contrast, ITSs are more complex. Second, prior research has not focused on using eye-tracking AOI data to model students' learning of representation skills. For example, Conati's research group used eye-tracking data in cognitive models found that it can improve predictions of students' learning of content knowledge [6]. This paper tests the hypothesis that eye-tracking AOI data improves cognitive models.

2. DATASET
We used data from a lab experiment that collected students' eye-tracking data while they worked with an ITS for chemistry for 3h [7]. 117 undergraduates participated in the experiment. For our analyses, we used log data from the ITS and eye-tracking data. To analyze the log data, we constructed a knowledge component (KC) model that relates each problem-solving step to the underlying skill. KCs corresponded to representation skills. To analyze the eye-tracking data, we generated visual attention features that assess how students process the visual representations with areas of interest (AOIs) that correspond to the representations. We also created AOIs for the parts of the screen where students solve problems, for the hint window, and for the periodic table that students could show and hide. We included only logged events and first attempts that were tagged with a KC with more than 30 data points. Our final dataset comprised a total of 30,893AOI and log events.

3. ANALYSES
We used two cognitive modeling approaches: Bayesian Knowledge Tracing (BKT) and Long Short Term Memory (LSTM) models. Both analyses used a 5 fold cross validation scheme which was created by assigning students to folds once.

BKT is the standard cognitive modeling procedure in research on ITSs [8]. We used BKT to evaluate a cognitive model representing performance prediction based on a student's history of incorrect and correct responses to questions of the same knowledge component. Following standard practice, we evaluated different guess and slip equivalence classes, which included using a different guess and slip per problem or per step. In previous work [9], separate guess and slip classes at the problem level resulted in a 10% gain in accuracy on ITS dataset. We applied this model to KCs without eye-tracking AOI data and to a version with eye-tracking AOI data. For the latter model, we fit a separate learning rate for each AOI within a problem.

All BKT models were fit with expectation maximization (EM) with max iteration of 100 and epsilon of 1e-6 as stop criteria. The best models in terms of log-likelihood used 40 EM restarts with initial parameter values. Tor prior these were drawn from a uniform random distribution, while the values for learn, guess, and slip were capped at 0.40, 0.40, and 0.30 respectively.

LSTM models are a subset of Recurrent Neural Networks (RNN). Recent progress in image classification with convolutional neural networks utilizes its ability to learn features that have more predictive power than manually crafted features (e.g., edge detection), previously the state of the art for image classification. In a similar vein, we used LSTM so that features of eye-tracking AOI data not yet known to be important could potentially be picked up. Therefore, the LSTM in represents a powerful detector to find out if there is a useful predictive signal in our sequences of eye-tracking AOI data.

We used two LSTM variants on RNNs that add a state to the hidden layer called the cell state which allows the network to

more effectively remember actions that occurred in the past when piecing together patterns in sequential input. We compared versions that utilized eye-tracking AOI data to versions that did not. Both LSTM models utilized the identical amount of information as their BKT with-eye and without-eye data counter parts and both trained a separate model per KC. In the case of LSTM models; eyeHeader, problemID-AOI, and Outcome comprised the feature vector. In both LSTM models, there is an instance of training data for every response given by a student. While non eye-tracking models were trained on sequence lengths that extend as long as the longest response sequence, AOI sequences were limited to the most recent N events, where N was defined as the maximum number of responses of any student in the training data + the median number of AOI events per student. This was done so that the data could fit into memory using 8bit signed integer matrices on a single large memory compute node.

4. RESULTS

After the 5 fold cross validation, RMSE was calculated per student. For a baseline reference, the RMSE of predicting the average percent correct for each KC was 0.39062. Models without eye-tracking data performed better than all of the models with eye-tracking data. Among the BKT models, problem was the better choice for assigning guess and slips over stepname, agreeing with prior work on ITS data [9]. Among LSTM models, extending the number of training epochs from 5 to 10 resulted in the most substantial gain of any model when not using eye-tracking but more epochs lead to overfit with the eye-tracking model. LSTMs, given the same problem-id and response data, were better able to leverage the information towards prediction accuracy than BKT, although both relied on a KC model. Differences between predictions were statistically reliable ($ps < 0.05$), as determined by a paired t-test of squared residuals between all adjacent models in the list with the exception of the LSTM model with 5 epochs and the BKT model with problem-id as guess/slip, which both used eye-tracking AOI data.

5. DISCUSSION

Our results stand in contrast to our hypothesis: using two cognitive modeling approaches, we did not find evidence that eye-tracking AOI data improves the accuracy of the model's prediction. This finding is noteworthy for the following reasons. First, it is counterintuitive because we tend to assume that visual attention is an important factor in assessing representation skills. Second, our finding stands in contrast to prior research on learning with text paired with one additional visual representation, where students view rather than interact with the material. The difference between prior work and our work is that our study used a complex learning environment, where students manipulated visual representations to solve problems. Third, our results stand in contrast to prior work, which found that eye-tracking AOI data can improve the accuracy of cognitive models of students' learning of content knowledge. The difference between prior work and our work is that our cognitive model assessed students' learning of representation skills, which reflects students' knowledge about the content and about visual representations.

One possible explanation is that prior eye-tracking research on learning with simple materials did not assess whether eye-tracking AOI data adds predictive accuracy to log data—because these materials do not generate log data. Second, representation skills may reflect not how students inspect visual representations, but how they use information from the representations to solve problems, which is sufficiently captured by the log data—particularly if the representations themselves are interactive and hence generate log data that can be used in cognitive models. Third, the fact that we modeled representation skills rather than content knowledge may explain why our results stand in contrast to prior work by Conati's group. We used a KC model that was specifically designed to assess students' representation skills. Even if eye-tracking AOI data assesses representation skills, it may simply not improve the accuracy of our cognitive model because the KC model already captures this information.

A limitation of our research results from the fact that the granularity of our AOIs was fairly coarse. Subtle cognitive signals may exist at fine grained resolutions which may require diving into the raw eye-tracking AOI coordinates. A second limitation was the exploration of hyper parameters. While this is always a caveat of any analysis using machine learning, a particular set of hyper parameters may exist which unlocks the predictive utility of the existing eye-tracking AOI data.

In sum, our findings suggest that eye-tracking AOI data does not necessarily add information relevant to students' representation skills, compared to what can be captured by a well-crafted KC model of representation skills. This rationale amounts to a new hypothesis that should be tested in future research: namely that adding representation skills to cognitive models of content knowledge may improve prediction accuracy in the same way as the addition of eye-tracking AOI data would.

6. ACKNOWLEDGMENTS

This work was supported by a grant from the National Science Foundation (IIS: BIGDATA 1547055).

7. REFERENCES

[1] Gilbert, J.K.: 'Visualization: An emergent field of practice and inquiry in science education': 'Visualization: Theory and practice in science education' (Springer, 2008), pp. 3-24

[2] NRC: 'Learning to Think Spatially' (National Academies Press, 2006)

[3] Koedinger, K.R., Corbett, A.: 'Cognitive Tutors: Technology bringing Learning Sciences to the classroom': 'The Cambridge Handbook of the Learning Sciences ' (Cambridge University Press, 2006), pp. 61-77

[4] Baker, R., Siemens, G.: 'Educational Data Mining and Learning Analytics': 'The Cambridge Handbook of the Learning Sciences' (Cambridge University Press, 2014), pp. 253-272

[5] Mason, L., Pluchino, P., Tornatora, M, Ariasi, N.: 'An eye-tracking study of learning from science text with concrete and abstract illustrations', The Journal of Experimental Education, 2013, 81, (3), pp. 356-384

[6] Bondareva, D., Conati, C., Feyzi-Behnagh, R., Harley, J.M., Azevedo, R., Bouchet, F.: 'Inferring learning from gaze data during interaction with an environment to support self-regulated learning': 'Artificial Intelligence in Education' (Springer, 2013), pp. 229-238

[7] Rau, M.A., Wu, S.: 'ITS support for conceptual and perceptual processes in learning with multiple graphical representations': 'Artificial Intelligence in Education' (Springer International Publishing, 2015), pp. 398–407

[8] Anderson, J.R., Boyle, C.F., Corbett, A.T., Lewis, M.W.: 'Cognitive modeling and intelligent tutoring' (Elsevier Science The MIT Press, 1990)

[9] Pardos, Z.A., Heffernan, N.T.: 'KT-IDEM: Introducing Item Difficulty to the Knowledge Tracing Model': 'User Modeling, Adaption and Personalization' (Springer, 2011), pp. 243-254

MATHia X: The Next Generation Cognitive Tutor

Steven Ritter
Stephen E. Fancsali
Carnegie Learning, Inc.
437 Grant Street, 20th Floor
Pittsburgh, PA 15219, USA
1.888.851.7094 {x122, x219}
{sritter, sfancsali}
@carnegielearning.com

ABSTRACT

MATHia X is the next generation implementation of Carnegie Learning's Cognitive Tutor (CT), a widely deployed, research-based mathematics curriculum that has provided data for many educational data mining studies. While many researchers are familiar with the basic operation of the system, there are several features that may affect analysis and interpretation of data that are less well known. We describe features of MATHia X and CT, as well as aspects of its practical implementation in real-world classrooms, that may be important for researchers using MATHia X and CT datasets.

Keywords

MATHia X, Cognitive Tutor, intelligent tutoring systems, real-world implementation, mastery learning, wheel-spinning

1. MATHIA X & COGNITIVE TUTOR

MATHia X is the next generation platform for Carnegie Learning's Cognitive Tutor (CT) [5], an intelligent tutoring system (ITS) for mathematics used by hundreds of thousands of learners in middle schools, high schools, and universities across the US (and to a lesser extent internationally, e.g., [4]).

MATHia X provides an HTML5/JavaScript, web-based implementation of the Cognitive Tutor technology and mathematics curricula; for our mid-2016 release we will have content for middle school grades 6-8 and Algebra I, with subsequent content covering Algebra II and Geometry. While MATHia X provides a technology and user interface refresh (including a space-themed interface "skin" in the initial release), fundamentally, most user interface and ITS affordances (including fine-grained data collected about learner interactions in the ITS) are essentially the same as they were in the Java-based Cognitive Tutor and MATHia products that have been in use for well over a decade. As such, we expect to continue in our long-standing tradition of partnering with education, educational data mining, and cognitive science researchers on basic and applied research about how students think and learn, as well as to continue providing data to these communities. The present demo explains a number of features common to both our legacy CT product as well as our next generation MATHia X product, many of which are important to data analyses carried by educational data mining researchers.

Datasets from CT are widely used in a variety of educational data mining (EDM) and education research projects, including in a substantial number of papers in the proceedings of the present conference. Many experimental and observational datasets (comprising hundreds of millions of learner actions in CT) have also been made available via the Pittsburgh Science of Learning Center's DataShop repository [3]. While many aspects of MATHia X and CT, such as their use of mastery learning and Bayesian Knowledge Tracing (BKT) are well known, there are many features and details of implementation and context of use that are less well known but important for appropriate analysis of CT (and eventually MATHia X) data. We describe a number of these characteristics here, in the hope that this information can inform EDM researchers' understanding of CT and MATHia X and contribute to future research that uses such data.

2. FEATURES & IMPLEMENTATION

2.1 Basal and Supplementary Use

Carnegie Learning produces text materials in addition to software, and the "blended" product (text and software) is often used as a "basal" curriculum, meaning that it is the primary source of instructional materials for a class. Our recommendation for blended implementations is that the software be used approximately 40% of the time (two class periods/week), with the text materials used for 60% of classroom time. Depending on school schedules, computer availability, and other factors, the amount of software usage varies considerably between schools.

In addition to "basal" usage, some schools use CT as a supplement to other educational materials. Such usage may follow the 60%-40% model, using a different textbook, but most supplemental usage is irregular. One consequence of such usage is that estimates of student knowledge can be highly inaccurate, since students may learn (or forget) substantial amounts in the long gaps between use of the tutor. Some supplementary use is for a specific purpose (e.g., summer school). In both types of implementations, schools may use the software for all students or for only a subpopulation thereof (e.g., those below grade level).

2.2 (Custom) Curricular Structure

Within K-12, there are a variety of main Carnegie Learning curricula: Algebra 1, Geometry and Algebra 2 (the high school sequence) are provided by our legacy CT product; a three-year middle school sequence and Algebra I are provided by the new MATHia X product in its initial release; and Bridge to Algebra, a one-year review of the middle school sequence is also provided on our legacy platform. Soon all of our curricula will be provided on the web-based technology that drives MATHia X. Overall, these curricula correspond to typical US courses. However, depending on state standards and other needs, schools may construct "custom" curricula that incorporate topics from one or more of these prototypical curricula. Custom curricula are popular, and the majority of CT data is now collected within such custom sequences. CT validates custom sequences for redundancies and violations of prerequisites; schools can ignore warnings about violations, but this is rare.

A curriculum consists of a set of modules, which represents a major topic in the curriculum. A full course may contain 6-8 modules. Modules consist of units, which consist of sections. Each section contains a large set of problems. Mastery learning

operates at the section level; students work within a section until they have mastered all associated knowledge components (KCs) (i.e., skills). The next section (or unit, if the section mastered is the final one in the unit) is automatically presented to the student. The module level is different. Although students will automatically progress to the next module when they complete the final section in the prior module, teachers can also "unlock" modules, allowing students to work on any open modules. Thus, at any given time, a student has a single position within a module (representing the current section) but may have positions within multiple modules. This feature is intended to allow movement among topics that do not have a prerequisite relationship.

2.3 Violations of Mastery Learning

Although we say CT and MATHia X implement mastery learning, in practice, there are several cases where students are not asked to work until they complete with mastery. Within each section of a curriculum, we specify a maximum number of problems that will be presented to students (often 25, but this varies, depending on the complexity of problems; for technical reasons, there are also cases where students might be promoted before reaching this maximum). If students complete this maximum without mastering their skills, they will advance to the next section of the curriculum. We call these advances "promotion," and these are flagged and communicated to teachers in our reporting system. The underlying idea is similar to the concept of "wheel-spinning" [1]. If students are not able to master the material in the tutor in a reasonable period of time, then it is likely that, for whatever reason, the tutor's mode of instruction for this topic is not resonating with the student, and so an alternate instructional approach is preferable. The teacher is responsible for presenting the alternative approach. Promotion is not rare; students are promoted from about 12% of sections. Promotions vary quite a bit by section and by student. Teachers also have the ability to manually move a student to a different position in the curriculum. Such placement changes also violate the mastery assumption. They happen for various reasons, most commonly because the teacher wants the student to "catch up" to the placement of the rest of the class. Such mastery learning violations due to placement changes are associated with greater error rates (and greater variability in error rates) over time than those experienced by students in classes that do not violate mastery learning [6].

2.4 Instructional Resources

Many analyses of CT data have looked at help seeking (e.g., [7]). Such work typically considers student use of problem-specific help, which is the only resource that affects CT's assessment of student knowledge. However, there are other sources of assistance available. Each unit has "lesson" content, which provides declarative instruction, worked examples, manipulatives, and topic-related video. A glossary is always available to students, and references to math terms within lesson text or hints are linked to it. Students also often use calculators and communicate with teachers and other students as they use the software.

Step-by-step examples provide another form of assistance. At least one example problem in each unit illustrates the basic problem-solving approach [2]. Unlike "regular" problems, step-by-step examples expose only one possible path through the problem, and text that would be used as a hint in problem solving is automatically presented to students as they go through the step-by-step example. This experience is intermediate between looking at a worked example and problem solving. Students can refer back to the step-by-step example as they work, and work in the step-by-step example is not used to assess student knowledge.

2.5 Non-persistent Student Model

Math knowledge is cumulative, so one expects that new topics incorporate many KCs mastered in earlier topics. Each section in CT and MATHia X monitors a small set of KCs, among the large set that is actually needed to solve problems in the section. While each section does introduce new knowledge, for various reasons, some sections list KCs that have been addressed in previous sections. These KCs take their preset values, not values based on students' prior work. In other words, CT and MATHia X do not assume that such KCs have been mastered. There is little practical consequence to listing such KCs; if the student learned them, CT will quickly recognize that fact, but researchers should be aware that the CT's assessment of skills is always within a section. Since skill values (i.e., estimates of student knowledge of a skill) do not carry over from section to section, researchers should not automatically assume that KCs with identical names in different sections are, in fact, identical KCs for purposes of data analysis.

3. DEMO + THE FUTURE

In this demo, we will exhibit basic problem solving in MATHia X, introducing the Cognitive Tutor technology to those unfamiliar with it and showing the refreshed technology to those already familiar with our products. Carnegie Learning looks forward to broad adoption of the next generation MATHia X software as a part of its blended mathematics curricula. Combining observational data sets from such adoptions with experimental data sets that will be collected by investigators using MATHia X as a platform for research will provide rich data to be mining and analyzed for many years to come in the educational data mining, learning analytics, cognitive science, and other research communities.

4. REFERENCES

[1] Beck, J.E., Gong, Y. 2013. Wheel-spinning: Students who fail to master a skill. In *Proceedings of AIED 2013* (Memphis, TN, Jul. 2013), 431-440.

[2] Hausmann, R.G.M., Ritter, S., Towle, B., Murray, R.C., Connelly, J. 2010. Incorporating interactive examples into the Cognitive Tutor. In *Proceedings of ITS 2010* (Pittsburgh, PA, Jun. 2010), 446.

[3] Koedinger, K.R., Baker, R.S.J.d., Cunningham, K., Skogsholm, A., Leber, B., Stamper, J. 2011. A data repository for the EDM community: the PSLC DataShop. In *Handbook of Educational Data Mining*, C. Romero, S. Ventura, M. Pechenizkiy, & R.S.J.d. Baker, Eds. CRC, Boca Raton, FL.

[4] Ogan, A., Walker, E., Baker, R., Rebolledo, G., Jimenez-Castro, M. 2012. Collaboration in Cognitive Tutor use in Latin America: Field study and design recommendations. In *Proceedings of CHI 2012* (Austin, TX, May 2012), 39-48.

[5] Ritter, S., Anderson, J.R., Koedinger, K.R., Corbett, A.T. 2007. Cognitive Tutor: applied research in mathematics education. *Psychon. B. Rev.* 14 (2007), 249-255.

[6] Ritter, S., Yudelson, M.V., Fancsali, S.E., Berman, S.R. 2016. How Mastery Learning Works at Scale. In *Proceedings of the 3rd Annual ACM Conference on Learning at Scale* (Edinburgh, Scotland).

[7] Roll, I., Aleven, V., McLaren, B.M., Koedinger, K.R. 2011. Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learn. Instr.* 21 (Apr. 2011), 267-280.

Towards Integrating Human and Automated Tutoring Systems

Steve Ritter, Stephen E. Fancsali, Susan Berman
Carnegie Learning, Inc.
437 Grant St.
Pittsburgh, PA 15219
{sritter, sfancsali, sberman}@carnegielearning.com

Michael Yudelson
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213
yudelson@cs.cmu.edu

ABSTRACT

We envision next generation learners having access to both automated and human sources of instruction in a variety of learning contexts. In such contexts, it will be most effective if students can be assisted to appropriately navigate between these sources of instruction. For example, human tutors, when helping a struggling student, might benefit from having access to the learning profile an automated tutor possesses on the student, including what the student already knows, detected misconceptions, inferred affective state and details about the student's work with the automated system before requesting human help. Similarly, an automated tutoring system would benefit from knowledge of interactions during human tutoring session. To facilitate student transitions between these types of systems, we need to understand the factors that best aid students in transitioning between such systems. This poster reports preliminary analyses, suggesting that students who are struggling with the course are more likely to take advantage of the optional human tutoring support and that such use is associated with increased course completion rates, regardless of the student's level of preparation.

Keywords

Human tutoring, intelligent tutoring system, blended approach.

1. INTRODUCTION

Intelligent tutoring systems (ITSs) frequently seek to mimic the best practices of one-on-one human tutors to drive improved student learning outcomes in a manner that is both scalable and cost effective. While extensive research considers a learning context in which a student uses an ITS while having a human instructor available (e.g., in K-12 computer labs), little work considers situations in which students use an automated tutoring system like an ITS alone (e.g., in their homes) while having human tutors available optionally for tutoring sessions via online chat. Data collected under such circumstances has the potential to generate important insight into how instructional "hand-offs" should proceed between such instructional modalities as well as general best practices for human and automated tutoring.

This project builds on more than a decade and a half of research on Carnegie Learning's Cognitive Tutor (CT) ITS [1]. The project leverages a unique dataset comprised of detailed learning records for thousands of students taking an online developmental math course. Students had required CT assignments as well as access to an online chat-based human tutoring service. This dataset allows us to explore the reasons that may lead students to choose to seek help from human tutors while using an intelligent tutoring system. The project also heavily draws on extensive work on tutorial dialogue data [2-3], allowing us to understand the human tutoring interactions that lead to the greatest learning gains within this context. At a technical level, the work further extends prior work exploring tutorial dialogue interactions and their automated classification by incorporating new and previously unavailable machine tutor data.

To the best of our knowledge, the proposed approach we are starting to work towards is the first attempt to address the creation and evaluation of an integrated approach to capitalize on the joint compensatory nature and data exchange between computerized tutors like ITSs and human tutors. We expect tools and results to generalize beyond the specific automated and human tutoring systems examined. For example, we expect knowledge gained from this work to inform us about how to better educate teachers about how to assist students in classrooms using the educational software in physical classrooms and how to build better reporting systems for human tutors helping students in a wide variety of educational applications.

As our first step in understanding how students navigate between CT and human tutoring (HT), we were particularly interested in understanding whether the subset of students who chose to use HT differed substantially in their use of CT and in their outcomes from students who did not use HT. In order to understand whether student preparation for the course affects use of HT, we use student performance in the first week of the course as a proxy for their initial ability in the course.

2. DATA

We collected data from two developmental college mathematics courses (one is a prerequisite for the other) deployed online at a degree-granting institution. Each course took place over five weeks, and the assignment for each week consisted of one large CT module. Each of these modules was broken into sections of content that grouped roughly similar problems. The instructional model within CT employs a mastery learning approach, in which, new problems are given until the CT's estimates of the underlying skills surpasses mastery thresholds. New sections of each math course begin every week; our dataset consists of all CT and HT interactions taking place from June 1 to December 31, 2014. The subject population consists of 16,905 CT users, approximately 3,300 of whom opted to request HT help during the selected period. These students produced over 19,000 human-tutored sessions, with an average length of 22 minutes. Students were predominantly adult learners of college age and older.

3. RESULTS

Table 1 shows primary descriptive statistics for these populations. Statistics for both courses were merged for simplicity since they are quite similar. The data indicate that students who opt to use HT struggled with the courses more than students who did not take advantage of HT. Students using HT have a higher assistance score (number of hints plus number of errors) in CT, as opposed to those who did not use HT. Perhaps as a result of asking for more hints and making more errors, students using HT worked more slowly, completing fewer sections per hour. The measure of sections per hour has been previously found to be predictive of overall course achievement [4].

These results are consistent with the idea that students who are struggling with the course are more likely to take advantage of HT. It seems unlikely that use of HT would have strong effects on course-level measures like amount of assistance or completion of sections per hour, since, on average, students who used HT used it fewer than 6 times in a course covering between 25 and 50 topics.

In contrast to these indicators that students using HT struggle with the course is the data showing that such students are more likely to complete sections in the course. That is, despite the fact that students turning to HT struggle with the course, they complete more sections of the course, indicating that HT may have a broad effect on student persistence.

To further investigate this effect, we use performance in the first module in the course as a proxy for students' initial preparation for the course. To better align Course 1 and Course 2, module 1 performance was converted to a z-score relative to the mean for that course and binned. Bin size was set to 0.5 standard deviations. Figure 1 shows means of course completion probability for each bin for users and non-users of HT with the number of students printed next to each point. At all levels of course preparation, students using HT, although, as we have seen, struggling, are more likely to complete the CT course material.

Table 1. CT and HT statistics: means (standard errors).

	Students using HT	Students not using HT
Parameter	**Course 1**	**Course 1**
CT sections attempted	50.25 (0.25)	50.25 (0.25)
CT problems attempted	493.22 (3.39)	359.38 (2.95)
CT assistance score	3003.16 (47.40)	2621.52 (47.70)
CT assistance score per section	62.78 (1.05)	71.36 (1.24)
CT time per student (hours)	35.41 (0.47)	35.85 (0.50)
CT sections mastered per hour	1.57 (0.03)	0.99 (0.02)
HT time per student (minutes)	110.05 (5.14)	N/A
HT utterances per student	352.82 (17.13)	N/A

4. Conclusion

These preliminary analyses provide a basis for understanding the factors that lead students to use HT and for understanding the broad influence of HT on students. These data are suggestive that students who are struggling with mathematics are more likely to use HT. Interestingly, the data are also suggestive that use of HT may have a broad affective influence on students. Despite the relatively small amount of contact with human tutors during the course, it appears that students who take advantage of such contact appear to be more willing to stick with the course and complete more work, despite their struggles with the mathematics.

5. ACKNOWLEDGMENTS

This work is supported by the contract with Advanced Distributed Learning agency of the Department of Defence (award W911QY-15-C-0070).

6. REFERENCES

[1] Ritter, S., Anderson, J.R., Koedinger, K.R., & Corbett, A. (2007). The Cognitive Tutor: Applied research in mathematics education. Psychonomic Bulletin & Review, 14(2), pp. 249-255.

[2] Morrison, D. M., Nye, B., & Hu, X. (2014). Where in the data stream are we?: Analyzing the flow of text in dialogue-based systems for learning. In R. A. Sottilare, X. Hu, H. Holden, & K. Brawner (Eds.) Design Recommendations for Intelligent Tutoring Systems: Volume 2: Adaptive Instructional Strategies and Tactics (pp. 217–223). U.S. Army Research Laboratory.

[3] Rus, V., D'Mello, S., Hu, X., & Graesser, A.C. (2013). Recent Advances in Conversational Intelligent Tutoring Systems, AI Magazine, 34(3):42-54.

[4] Ritter, S., Joshi, A., Fancsali, S.E., and Nixon, T. (2013). Predicting Standardized Test Scores from Cognitive Tutor Interactions. In Proc. of the 6th International Conf. on Educational Data Mining (Memphis, TN, July 6-9, 2013). 169-176.

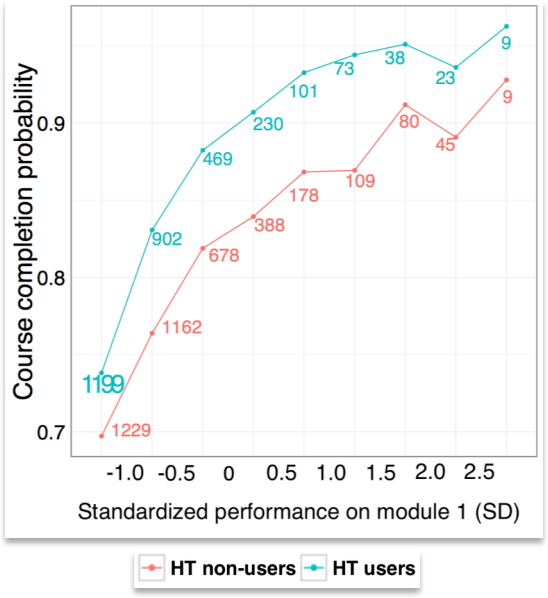


Figure 1. Standardized performance on module 1 vs. overall course completion probability.

Toward Revision-Sensitive Feedback in Automated Writing Evaluation

Rod D. Roscoe
Arizona State University
Rod.Roscoe@asu.edu

Matthew E. Jacovina
Arizona State University
Matthew.Jacovina@asu.edu

Laura K. Allen
Arizona State University
LauraKAllen@asu.edu

Adam C. Johnson
Arizona State University
acjohn17@asu.edu

Danielle S. McNamara
Arizona State University
Danielle.McNamara@asu.edu

ABSTRACT

Revising is an essential writing process yet automated writing evaluation systems tend to give feedback on discrete essay drafts rather than changes across drafts. We explore the feasibility of automated revision detection and its potential to guide feedback. Relationships between revising behaviors and linguistic features of students' essays are discussed.

Keywords

Automated Writing Evaluation; Writing; Revising; Intelligent Tutoring Systems; Natural Language Processing; Feedback

1. INTRODUCTION

Automated writing evaluation (AWE) systems provide computer-based scores and feedback on students' writing, and can promote modest gains in writing quality [1, 2]. One concern is that students receive feedback on their *current* drafts that ignores *patterns of change* from draft to draft. We argue AWE tools should include feedback models that incorporate data on students' revising behaviors and textual changes. These innovations may afford greater personalization of formative feedback that helps students recognize how their editing actions affect writing quality.

This study used Writing Pal (W-Pal), a tutoring and AWE system that supports writing instruction and practice [3, 4]. When submitting essays to W-Pal, students receive scores (6-point scale) and feedback with actionable suggestions for improvement. Scoring and feedback are driven by natural language processing (NLP) algorithms that evaluate lexical, syntactic, semantic, and rhetorical text features [1, 5]. One goal for W-Pal development is feedback that promotes more effective revising [see 4].

2. METHOD

2.1 Context and Corpus

High school students (n = 85) used W-Pal to write persuasive essays on the topic of "fame." Most identified as native English speakers (56%) and others as English-language learners (44%).

2.2 Detection and Annotation of Revising

We calculated difference scores between drafts for several NLP measures (via Coh-Metrix [5, 6]). Lexical measures assessed word choice and vocabulary, such as word frequency and hypernymy. Cohesion indices assessed factors such as overall essay cohesion, semantic relatedness (using LSA), and structure.

Human annotation of revisions adapted methods from prior research [7, 8]. Writers can alter their text via adding, deleting, substituting, or reorganizing actions. Human coding of these revision actions showed high reliability (κ = .92). Revisions can also maintain (superficial edits) or transform (substantive edits) the meaning of surrounding text. Human coding of revision impact on text meaning also demonstrated high reliability (κ = .81).

3. RESULTS

3.1 Automated Detection of Revising

Essays demonstrated detectable changes in linguistic features from original to revised drafts. Revised essays were longer, included more transitional phrases and first-person pronouns, and were somewhat more cohesive (see Table 1).

Table 1. Linguistic Changes and Correlations with Scores

	Linguistic Change		Correlation with Score Change	
Linguistic Change	***t*(84)**	***p***	***r*(84)**	***p***
Basic				
Word Count	**6.24**	**< .001**	.06	.593
Sentence Count	**4.33**	**< .001**	-.09	.393
Lexical				
Lexical Diversity	-0.28	.781	.17	.124
Word Concreteness	0.83	.410	**.34**	**.002**
Word Familiarity	-0.74	.463	-.01	.954
Word Hypernymy	0.80	.424	**.24**	**.028**
1st Person	**2.09**	**.040**	-.07	.545
2nd Person	-1.06	.294	**-.22**	**.043**
3rd Person	-0.23	.818	-.10	.342
Cohesion				
Connectives	**1.67**	**.099**	.03	.809
LSA Given/New	**2.98**	**.004**	.08	.484
LSA Sentences	0.58	.562	**.24**	**.029**
LSA Paragraphs	**1.86**	**.066**	-.08	.465
Deep Cohesion	0.71	.478	**.18**	**.098**
Referential Cohesion	0.52	.607	.01	.893
Narrativity	1.05	.296	**-.25**	**.023**

Essay quality increased from original (M = 2.7, SD = 1.0) to revised drafts (M = 2.9, SD = 1.1), $t(84) = 3.64$, $p < .001$, $d = .19$. Gains correlated with increased concreteness, specificity, objectivity (i.e., fewer 2nd-person pronouns and less story-like), and cohesion. Importantly, the linguistic changes linked to gains were *not* the most typical changes. This finding reinforces the idea that students are not skilled revisers—their revising behaviors can be dissociated from actions that improve the quality of their work.

3.2 Human Annotation of Revising

The most common revisions were additions (47.5%) and substitutions (33.6%). Deletions (15.4%) and reorganizations (2.5%) occurred less often. None of the revising actions were correlated with changes in essay score. This finding reiterates the point that high school students are not necessarily skilled revisers.

3.3 Relationships between Modes of Analysis

The total number of revisions was not related to linguistic changes across drafts (range of *r*s from -.18 to .12). Simply revising *more* had minimal effects. Additions, substitutions, and reorganization had few effects. In contrast, deletions were associated with reductions in narrativity and third-person pronouns. Along with reduced word familiarity, this pattern suggests that students were removing story-like language. Deletions were also associated with reduced given information, semantic similarity across paragraphs, and referential cohesion. Thus, as students removed content from their essays, the cohesive flow of ideas was perhaps hindered. Overall, deletions seemed to be linked to both gains and setbacks in essay quality (see Table 2).

Table 2. Correlations of Revision Types and Linguistic Change

Linguistic Change	Add	Delete	Subst.	Reorg.
Basic				
Word Count	**.29**[b]	**-.36**[a]	-.18	-.10
Sentence Count	**.37**[a]	-.18	-.16	.05
Lexical				
Lexical Diversity	.01	**.26**[c]	-.04	.07
Word Concreteness	.00	**.29**[b]	.08	.06
Word Familiarity	-.04	**-.28**[c]	.15	-.09
Word Hypernmy	-.10	.11	.02	-.18
1st Person	.04	-.11	.11	.07
2nd Person	-.09	-.03	-.05	-.04
3rd Person	-.01	**-.26**[c]	-.07	.00
Cohesion				
Connectives	-.07	.16	.09	-.03
LSA Given/New	-.02	**-.32**[c]	-.07	-.07
LSA Sentences	-.20	-.09	.06	-.12
LSA Paragraphs	.07	**-.24**[c]	-.05	.04
Deep Cohesion	.00	-.11	.07	-.07
Referential Cohesion	-.10	**-.25**[c]	.12	-.03
Narrativity	-.07	**-.34**[a]	-.01	.01

Note. [a]$p \leq .001$. [b]$p \leq .01$. [c]$p \leq .05$.

A final analysis examined revisions by both type *and* impact. As in the previous analysis, the most meaningful linguistic changes were associated with deletions, with substantive deletions appearing to have the strongest influence. Superficial deletions tended to make essays more personalized (i.e., more 1st-person pronouns) and less specific. Substantive deletions tended to make essays shorter, less story-like, more sophisticated in terms of vocabulary, and less cohesive.

4. Discussion

Our results provide evidence that automated tools can detect linguistic changes in students' writing. Formative feedback based on such measures might help students appreciate when and how their drafts evolve over time. For instance, when an increase in narrativity or decrease in cohesion are detected, feedback could flag the edited sections of text so that conscientious students can draw inferences about the impact of their revisions.

Ideally, AWEs should also be able to detect and give feedback on revising behaviors. From the current study, however, it is unclear whether linguistic data could be used to identify such behaviors. With the exception of deletions, students' revising actions did not have a profound impact on linguistic properties.

One solution may reside in keystroke logging [9]. Keyboard and mouse clicks made while interacting with an AWE system may be interpretable with respect to revising. For example, backspace presses may indicate deletion. The use of mouse buttons to select text, along with "CTRL-X" and "CTRL-V" hotkey functions, could signal reorganization. If such tools can be added to AWEs, they may provide real-time measures of writing and revising behaviors that can be explicitly linked to linguistic consequences.

5. ACKNOWLEDGMENTS

This research was supported in part by the Institute for Educational Sciences (IES R305A120707). Opinions, findings, and conclusions or recommendations expressed are those of the authors and do not necessarily reflect the views of the IES.

6. REFERENCES

[1] Shermis, M., and Burstein, J. C. (Eds). 2013. *Handbook of automated essay evaluation: current applications and new directions.* Routledge.

[2] Stevenson, M., and Phakiti, A. 2013. The effects of computer-generated feedback on the quality of writing. *Assessing Writing*, *19*, 51-65.

[3] Roscoe, R. D., Allen, L. K., Weston, J. L., Crossley, S. A., and McNamara, D. S. 2014. The Writing Pal intelligent tutoring system: usability testing and development. *Computers and Composition*, *34*, 39-59.

[4] Roscoe, R. D., Snow, E. L., Allen, L. K., and McNamara, D. S. 2015. Automated detection of essay revising patterns: applications for intelligent feedback in a writing tutor. *Technology, Instruction, Cognition, and Learning*, *10*, 59-79.

[5] McNamara, D. S., Crossley, S. A., Roscoe, R. D., Allen, L. K., and Dai, J. 2015. Hierarchical classification approach to automated essay scoring. *Assessing Writing*, *23*, 35-59.

[6] McNamara, D. S., Graesser, A. C., McCarthy, P., and Cai, Z. 2014. *Automated evaluation of text and discourse with Coh-Metrix*. Cambridge University Press.

[7] Faigley, L., and Witte, S. 1981. Analyzing revision. *College Composition and Communication*, *32*, 400-414.

[8] Fitzgerald. J. 1987. Research on revision in writing. *Review of Educational Research*, *57*, 481-506.

[9] Leijten, M., and Van Waes. 2013. Keystroke logging in writing research: using Inputlog to analyze and visualize writing processes. *Written Communication*, *30*, 358-392.

Preliminary Results on Dialogue Act Classification in Chat-based Online Tutorial Dialogues

Vasile Rus, Rajendra Banjade
Department of Computer Science
The University of Memphis
Memphis, TN 38152
{vrus,rbanjade}@memphis.edu

Nabin Maharjan, Donald Morrison
The University of Memphis
Memphis, TN 38152
{nmharjan}@memphis.edu

Steve Ritter, Michael Yudelson
Carnegie Learning, Inc.
437 Grant Street, Suite 918
Pittsburgh, PA 15219, USA
{sritter}@carnegielearning.com

ABSTRACT

We present in this paper preliminary results with dialogue act classification in human-to-human tutorial dialogues. Dialogue acts are ways to characterize the intentions and actions of the speakers in dialogues based on the language-as-action theory. This work serves our larger goal of identifying patterns of tutors' actions, in the form of dialogue act and subact sequences, that relate to various aspects of learning. The preliminary results we obtained for dialogue act classification using a supervised machine learning approach are promising.

Keywords

dialogue acts, intelligent tutoring systems, instructional strategies.

1. INTRODUCTION

A key research question in intelligent tutoring systems and in the broader instructional research community is understanding what expert tutors do. A typical operationalization of this goal of understanding what expert tutors do is to define the behavior of tutors based on their actions.

In our case, because the focus is tutorial dialogues, we model the actions of tutors using dialogue acts inspired from the *language-as-action* theory [1, 7]. According to the language-as-action theory, *when we say something we do something*. Therefore, we map all utterances in a tutorial dialogue onto corresponding dialogue acts using a predefined dialogue act taxonomy, which is described later. It should be noted that automatically discovered dialogue act taxonomies are currently being built [6]. However, we chose to work with an expert-defined taxonomy of dialogue acts, developed by experts based on dialogue and pedagogical theories [5], because it better serves our larger research goals of testing such theories.

2. THE APPROACH

We adopted a supervised machine learning method to automate the process of dialogue act classification. This implies the design of a feature set which can then be used together with various supervised machine learning algorithms such as Naive Bayes, Decision Trees, and Bayes Nets. For automated dialogue act classification, researchers have considered rich feature sets that include the actual words (possibly lemmatized or stemmed) and n-grams (sequences of consecutive words). Besides the computational challenges posed by such feature-rich methods, it is not clear whether there is need for so many features to solve the problem of dialogue act classification.

Our approach is based on the observation that humans infer speakers' intention after hearing only a few of the leading words of an utterance [4]. One argument in favor of this assumption is the evidence that hearers start responding immediately (within milliseconds) or sometimes before speakers finish their utterances ([5] - pp.814).

Intuitively, the first few words of a dialog utterance are very informative of that utterance's dialogue act. We could even show that some categories follow certain patterns. For instance, Questions usually begin with a *Wh*-word while dialogue acts such as Greetings use a relatively small bag of frozen words and expressions.

In the case of other dialogue act categories, distinguishing the dialogue act after just the first few words is not trivial, but possible. It should be noted that in typed dialogue, which is a variation of spoken dialogue, some information is not directly available. For instance, humans use spoken indicators such as the intonation to identify the dialogue act of a spoken utterance. We must also recognize that the indicators allowing humans to classify dialogue acts also include the expectations created by previous dialogue acts, which are discourse patterns learned naturally. For instance, after a first Greeting another Greeting that replies to the first one is more likely. We used intonational clues in our work to the extent that such information is indirectly available to us, in the form of punctuation marks, in typed/chat-based dialogues. We did incorporate contextual clues in our preliminary experiments, e.g. we used as a feature the dialogue act of the previous utterance, but the results did not improve significantly. It is important to note that the present study assumes there is one direct speech act per utterance.

3. THE TAXONOMY

The current coding taxonomy builds on an earlier taxonomy that sought to identify patterns of language use in a large corpus of online tutoring sessions conducted by human tutors in the domains of Algebra and Physics [5]. The taxonomy is considerably more granular than previous schemes such as the one used by Boyer and colleagues [2].

The most recent version of the taxonomy employs two levels of description. At the top level, it identifies 16 standard dialogue categories including Questions, Answers, Assertions, Clarifications, Confirmations, Corrections, Directives, Explanations, Promises, Suggestions, and so forth. It also includes two categories, Prompts and Hints, that have particular pedagogical purposes. Within each of these major dialogue act categories we identify between 4 and 22 subcategories or subacts.

4. EXPERIMENTS AND RESULTS

We have used in our experiments 288 tutorial sessions (containing about 17,537 utterances) between professional human tutors and actual college-level, adult students. These sessions are a subset of of a larger sample of 500 sessions randomly selected from a corpus of 17,711 sessions we obtained from an organization that offers online human tutoring services. Students taking two college-level developmental mathematics courses (pre-Algebra and Algebra) were offered these online human tutoring services at no cost. The same students had access to computer-based tutoring sessions through Adaptive Math Practice, a variant of Carnegie Learning' Cognitive Tutor. It should be noted that students may or may not initiate a tutorial dialogue with a human tutor while attending those courses. This is important to note as there could be a self-selection bias in those tutorial dialogues that we used.

Expert Annotation Process

The 288 sessions we used here were manually labelled by a team of 6 trained annotators, all of whom were experienced classroom math teachers. Each session was first manually tagged by two independent annotators, i.e. they did not see each other's tags. Then, the tags of the two independent annotators were double-checked by a verifier, who also happens to be the designer of the taxonomy. The verifier had full access to the tags assigned by the independent taggers. The role of the verifier was to resolve discrepancies. The inter-annotator agreement for the two independent annotators was Cohen's kappa=0.72 for dialogue acts and kappa=0.60 for dialogue acts and subacts combined.

The agreement was best for Expressives (0.88), Assertions (0.81), Requests (0.78) and worst for Hints (0.2), Clarifications (0.33), and Explanations (0.42).

Results

For space reasons, we summarize the results of our supervised machine learning approach in terms of accuracy and Cohen's kappa relative to the final tag adjudicated by the verifier using a 10-fold cross-validation approach. We only provide results on dialogue act classification (no subacts) for the same space reasons.

The model

Our model for predicting dialogue acts consists of the following five features/predictors: the leading three tokens in an utterance, the last token such as a question mark ('?') at the end of a question, and the length of the utterance. We experimented with other features such as the speaker (student vs. tutor), the position of the utterance in the dialogue, e.g. an utterance at the beginning of a session is more likely a Greeting, the previous dialogue act, but we have not noticed any significant impact on performance relative to the five-feature model mentioned above. More powerful models that do account explicitly for sequential observations are needed, e.g. Conditional Random Fields.

We experimented with our 5-feature model in combination with a number of machine learning algorithms including Naïve Bayes, Decision Trees, and Bayes Nets. We also experimented with sequential models based on Conditional Random Fields but the results, again, were not better. The best results, obtained with BayesNets, are summarized below.

D-Act classification Results

Using all features leads to 67.27% accuracy and Cohen's kappa of 0.58. The speaker does not seem to have an impact as the results accuracy is 66.74%. The same for position, if removed the resulting accuracy is 66.77%. The remaining features are indeed important as if another is removed the accuracy drops significantly below 60.00%.

Our plan next is to annotate more sessions up to 500 and retrain our models. Once the accuracy is at acceptable level, we will use the classifiers to automatically tag tens of thousands of sessions with dialogue acts and subacts. Once the sequences of actions and subactions are available, we will identify patterns of tutor and student actions that related to learning and affect and which could then be used in the development of automated intelligent tutoring systems or in a hybrid system where both human and intelligent tutors co-exist.

Acknowledgments. This research was sponsored by a subcontract to The University of Memphis from Carnegie Learning, Inc., under award W911QY-15-C-0070 by Department of Defense. Any opinions, findings, and conclusions or recommendations expressed in this material are solely the authors'.

5. REFERENCES

[1] Austin, J. L. (1962). *How to do things with words*: Oxford University Press, 1962.

[2] Boyer, K.E., Phillips, R., Ingram, A., Ha, E.Y., Wallis, M.D., Vouk, M.A., & Lester, J.C. (2011). Investigating the relationship between dialogue structure and tutoring effectiveness: a hidden Markov modeling approach, The *International Journal of Artificial Intelligence in Education (IJAIED)*, Vol. 21 No. 1, 2011, 65-81.

[3] Jurafsky, Dan.; and Martin, J.H. (2009). Speech and Language Processing. Prentice Hall, 2009.

[4] Moldovan, C., Rus, V., & Graesser, A.C. (2011). *Automated Speech Act Classification for Online Chat*, The 22nd Midwest Artificial Intelligence and Cognitive Science Conference, Cincinatti, OH, April 2011 (Best Student Paper Award - Honorary Mention).

[5] Morrison, D. M., Nye, B., Samei, B., Datla, V. V., Kelly, C., & Rus, V. (2014). *Building an Intelligent PAL from the Tutor.com Session Database-Phase 1: Data Mining*. The 7th International Conference on Educational Data Mining, 335-336.

[6] Rus, V., Graesser, A., Moldovan, C., & Niraula, N. (2012). *Automatic Discovery of Speech Act Categories in Educational Games*, 5th International Conference on Educational Data Mining (EDM12), June 19-21, Chania, Greece.

[7] Searle, J. R. (1969). *Dialogue Acts: An essay in the philosophy of language*. Cambridge university press, 1969.

SAS Tools for Educational Data Mining

Jennifer Sabourin
SAS Institute
100 SAS Campus Dr.
Cary, NC 27513
1. 919.531.3313
Jennifer.Sabourin@sas.com

Scott McQuiggan
SAS Institute
100 SAS Campus Dr.
Cary, NC 27513
1. 919.531.1119
Scott.McQuiggan@sas.com

Andre de Waal
SAS Institute
100 SAS Campus Dr.
Cary, NC 27513
1. 919.531.6575
Andre.Dewaal@sas.com

ABSTRACT

Researchers in the EDM community have always relied on sophisticated tools to analyze data and build models. As the amount of data that can be collected and stored grows, the need for tools capable of handling "big data" becomes ever more prevalent. SAS® Analytics U is a new initiative for making SAS data analysis and mining tools available for free to educational researchers and instructors. These tools are designed for handling very large data sets and can be run in the cloud, saving researchers valuable time and resources. Furthermore, SAS Analytics U provides a community of SAS educators and learners to share resources and information about SAS tools and techniques.

This tutorial aims to introduce researchers to the tools available through SAS Analytics U and how they can be applied to the field of Educational Data Mining. We will provide an overview of the SAS architecture and provide instruction on the key features of each tool in the suite. We will guide participants through examples using relevant educational data sources to help researchers understand how the tools can be applied to their own work.

REQUIREMENTS: In order to participate in the hands on exercises, please bring a laptop on which you have installed SAS University Edition. The free download is available at http://www.sas.com/en_us/software/university-edition/download-software.html. The download and installation may take up to 1 hour so there will not be time to get set up during the tutorial.

1. TUTORIAL DESCRIPTION

This tutorial will focus on introducing SAS to participants and guiding them through the use of the suite of tools using relevant educational data sets. The tools that will be covered include:

SAS® Programming Language. SAS programming language is a powerful language designed specifically for intensive data analysis. This highly flexible and extensible fourth generation programming language has a clear syntax and hundreds of language elements and functions. It supports programming everything from data extraction, formatting and cleansing to data analysis, building sophisticated models, and generating reports. The SAS programming language is at the heart of the SAS University Edition tools.

SAS® Studio. SAS Studio is the development environment for SAS University Edition and runs through the web browser as well as in the cloud. It offers a powerful GUI interface that allows novice programmers to interact with data and perform analyses without writing any SAS code themselves. However, the SAS code is all generated behind the scenes and is visible to help users learn.

SAS® Enterprise Miner. SAS Enterprise Miner helps users streamline the data mining process to create highly accurate predictive and descriptive models based on analysis of vast amounts of data. It includes innovative algorithms in the areas of statistics and machine learning to enhance the stability and accuracy of predictions, which can be verified easily by visual model assessment and validation. Users build process flow diagrams that serve as self-documenting procedures. These diagrams can be updated easily or applied to new problems without starting over from scratch. In addition to process flow diagrams, Enterprise Miner provides a programming interface for advanced users. Enterprise Miner allows integration with open source software for data manipulation and model comparison, the open standard PMML, and databases for scoring models without data movement.

Additional SAS tools that may be covered if it is of interest to the participants include tools for time series analysis, forecasting, matrix manipulations, and advanced statistics.

2. JUSTIFICATION

Educational data miners rely on computational tools to understand and explore their data. These tools must be robust and flexible in order to allow for innovation. They must be able to handle ever increasing amounts of data. Ideally, they are easy to use by both programmers and non-programmers alike due to the interdisciplinary nature of this research area. Finally, most researchers rely upon tools that are freely available and do not require excessive resources.

SAS University Edition is a new option that addresses many of these needs. This suite of powerful SAS software was made available to all learners for free in May of 2014. SAS Enterprise Miner, Text Miner, and Forecast Server have been available through SAS OnDemand for Academics since late 2010. However, the biggest barrier to adopting new tools is learning how to use them. SAS Analytics U is a community centered around these free offerings and is designed to support SAS learners and educators. This tutorial seeks to introduce participants to these resources and suite of tools and demonstrate how they can be applied to EDM research. The goal is that participants will be able to add another set of tools to their every growing toolbox for conducting EDM research.

3. PRESENTERS

The presenters for this tutorial include both researchers who are active in the EDM community and trained SAS educators who are experienced in leading tutorials of SAS products.

Jennifer Sabourin. Sabourin has a dual role as a research scientist and software developer on the Curriculum Pathways team at SAS Institute. As a research scientist she works on identifying research questions and using machine learning and analytical techniques to improve the efficacy of Curriculum Pathways products. She also serves as a consultant aiding external researchers with using SAS

software to better understand and make decisions from their educational data. As a software developer she works on creating innovative applications for K-12 that are offered at no-cost.

Sabourin received her Ph.D. from North Carolina State University in 2013. Her graduate work focused on data mining and artificial intelligence in game-based learning environments. She has been an active member of the EDM community since beginning her graduate work.

Scott McQuiggan. McQuiggan leads SAS Curriculum Pathways, an interdisciplinary team focused on the development of no-cost educational software in the core disciplines at SAS Institute Inc. Curriculum Pathways includes more than 1,500 resources, tools, and apps for K-12 education used in all 50 states and more than 90 countries around the world. He regularly uses data mining and analytics to better understand the behaviors exhibited in Curriculum Pathways resources and improve the efficacy of the products themselves.

McQuiggan received his PhD in computer science from North Carolina State University, where his research focused on affective reasoning in intelligent game-based learning environments. He also holds an MS in computer science from North Carolina State University and a Bachelor of Science in computer science from Susquehanna University. Scott is co-author of the book, Mobile Learning: A Handbook for Developers, Educators, and Learners.

André de Waal. De Waal is an Analytical Consultant with SAS Institute and his work focuses on teaching users how they can use SAS to best meet their analytic needs. He received his Ph.D. in theoretical computer science from the University of Bristol during 1994. He spent the next year in Germany and Belgium continuing his research in Logic Programming and Automated Theorem Proving. During 1996 he returned to South Africa to take up his position as lecturer at the School of Computer Science and Information Systems at the then Potchefstroom University for Christian Higher Education (which later became the North-West University), where he was later promoted to Associated Professor. During 1999 he became one of the founder members of the Centre for Business Mathematics and Informatics at the same university. He became responsible for the Data Mining Program in the Centre and shifted his research focus to include Neural Networks and Predictive Modeling. He joined SAS Institute in Cary, NC during December 2010 to take up the position of Analytical Consultant in the Global Academic Program.

4. PROPOSED FORMAT

This tutorial will be presented as interactive instructions where users will be guided through the tools using relevant education data with a focus on techniques that are commonly required in the EDM community. The tutorial will also include an overview of SAS and its commitment to education research by a leading SAS executive. We also seek to gain feedback from participants prior to the event so that we can tailor the sessions to specific needs or questions. A tentative schedule (subject to conference timings) is below:

Session 1: Introduction and SAS Studio

9:00-9:15	Introduction – Introduction of presenters and participants and overview of SAS Analytics U
9:15-10:30	SAS Studio
Coffee Break	

Session 2: SAS Studio

11:00-12:30	SAS Studio
Lunch Break	

Session 3: Keynote and SAS Enterprise Miner

14:00-14:30	Keynote – A SAS executive (TBD based on final scheduling) will present an overview of SAS and its commitment to education by discussing tools made available to researchers and products made available to K-12 educators and students.
14:30-16:00	SAS Enterprise Miner
Coffee Break	

Session 4: Participant Requested Instruction

16:30-17:30	Additional Instruction – based on the goals of the participants we will delve deeper into aspects of the tools already presented or introduce additional tools as listed in the tutorial description.
17:30-18:00	Conclusion

In addition to the tutorial, instructional materials will be made available to participants. We will also provide guidance on avenues for further learning through online instruction.

Applicability of Educational Data Mining in Afghanistan: Opportunities and Challenges

Abdul Rahman Sherzad
PhD Student at Technische Universität
Berlin, Germany
absherzad@gmail.com

ABSTRACT

The author's own experience as a student and later as an active lecturer in Afghanistan has shown that the methods used in the Afghan educational systems do not provide students with the minimum guidance needed to select the proper course of study before they enter the national university entrance exam (Kankor). The result is often high attrition rates and poor performance in higher education.

Based on the studies done in other countries, and by the author of this paper through online questionnaires distributed to university students and graduates in Herat, Afghanistan – it was found that proper procedures and specialized studies in high schools can help students in selecting their field of study more systematically. Additionally, there are large amounts of data available for mining purposes but the methods that the Ministry of Education and Ministry of Higher Education use to store and produce their data only enable them to achieve simple facts and figures. Furthermore, from the results it can be concluded that there are potential opportunities for educational data mining application in the domain of Afghanistan's education systems. For instance, predict proper field of study for high school graduates, or, identify first year university students who are at high risk of attrition.

Keywords

Educational data mining; major prediction; student placement; Kankor; Afghanistan education systems; value of information.

1. INTRODUCTION

General education in Afghanistan comprises K-12 (primary, secondary and high school), Islamic studies, Teacher Training, Technical and Vocational schools and institutes which are administered by the Ministry of Education (MoE). The Ministry of Higher Education (MoHE) supervises universities which provide Bachelor's, Master's, and PhD degree programs.

Since the establishment of the new democracy in Afghanistan in 2001, education systems have been going through a nationwide rebuilding process. Despite obstacles, numerous public and private educational institutions were established across the country [2]. The result is a substantial increase in the student enrollment rate, as reflected (see Figure 1).

Every year more than 200,000 students graduate from high schools and around 300,000 participate in Kankor across the country [3].

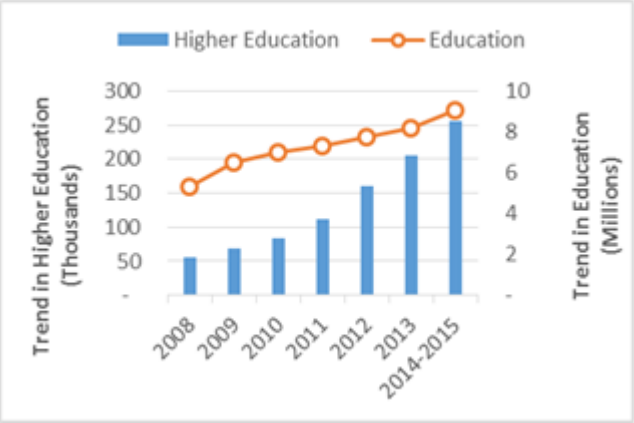


Figure 1. Education and Higher Education enrollment trends.

The MoE and MoHE as the main bodies of education systems in Afghanistan have been trying to standardize the quality of education in order to be able to meet the minimum international standards. In this extremely challenging process, one of the efforts of the MoE and MoHE has been to automate their information through Education Management Information System (EMIS) and Higher Education Management Information System (HEMIS) [6].

The EMIS and HEMIS are able to generate (only) basic statistics (e.g., total number of students and teachers based on gender, geographic location, schools and universities) which are not very helpful in decision making to improve the education systems effectively. For example, '10 million students in schools' is just a number and piece of data without a specific context and further useful information to describe the setting. Hence, these simple facts and figures do not help policy makers to improve the educational settings. For example, one cannot predict proper majors/fields of study (Major) for high school graduates, or, identify first year university students who are at high risk of attrition. This paper will be a new initiative in its kind. The objective is to study the opportunities and challenges of EDM applicability in the Afghanistan education context in order to help educational institutions to better prepare students for their studies in schools and universities.

2. MAJOR RECOMMENDATION

Presently in Afghanistan, school students are not divided into Majors. The author conducted one online survey to public and private university students and graduates, and another survey to computer science students and graduates. A total of 333 people participated in these surveys; 315 agreed that it is more useful if the students are offered specialized studies after grade 9 at school.

Additionally, due to general studies and insufficient orientation on Kankor at schools, the majority of students do not know what Major to choose in the Kankor. This was confirmed by the same online surveys. Besides, in the existing situation, it is found that there are no structural and specialized institutions to provide and guide students on career choices based on their skills and interests. This situation creates a vicious cycle for misappropriating human-capital as the most vital resource for development.

The outcome of these studies [4, 7] can be customized and used to recommend proper Majors to high school graduates prior attending the Kankor, and also while specialized studies are introduced at schools. The following approaches can be used. 1-Assess student performance for 10th, 11th and 12th grades to identify the strengths and weaknesses of the applicants in all the relevant Majors. 2-Since the results of high school grades could be misleading, this research proposes the design of a new standardized test in order to evaluate the interest and capabilities of the applicants through varied 'Yes' and 'No' intelligent questions. 3-Since there are no pre-collegiate courses prior to entering University, it is deemed efficient to evaluate the skills of applicants in the 12th grade through a number of Kankor practice tests. 4-Other simulator (self-assessment) tools as an all-encompassing medium to self-evaluate, capitalize on improving and minimize the identified gaps of candidates and to evaluate the interest and capabilities of the applicants. 5-Of course, social, economic, and literacy status of student's family and other pedagogical factors could be significant for better evaluation and assessment. 6-Divide more than 100 Majors into main major areas including Natural and Social Sciences, Health Sciences, Humanities and Literature, Islamic Education, Fine Arts and Technical Education. 7-Last but not least, consideration of previous Kankor results data during data mining process would lead to better accuracy rate.

3. SUPPORT AT RISK STUDENTS

Most of the students are at risk of dropping out or performing poorly during their higher education studies. One of the main reasons is that the participants randomly select Majors in the Kankor without much knowledge of the requirements and challenges ahead of them and the inventory of their existing knowledge in the relevant field of study. Also, lack of specialized studies at schools is another major reason for attrition and poor performance in higher education. According to the above mentioned online survey conducted by the author among Computer Science students in Herat province out of 227 respondents around 90% did not have the skills and knowledge of basic programming, database, and operating systems, as echoed in (see Figure 2). The result of the survey is showing that one of the major reasons for weak academic performance in higher education is lack of specialized studies in school.

An early counseling intervention solution would be a great support to identify the key factors to improve their academic performance and to decrease rates of attrition through academic counseling, tutorial classes and other supportive programs [1, 5]. This could be achieved with evaluation and comparison of fresh student's data with historical data of senior students. For example, school performance and grades for main prerequisite subjects relevant to their selected Major (i.e. the required score value for Journalism in mathematics might be 2 out of 5, while in Engineering it might be, 5 out of 5), if they attended supportive courses and classes besides school studies, family responsibilities, and other social and extracurricular activities.

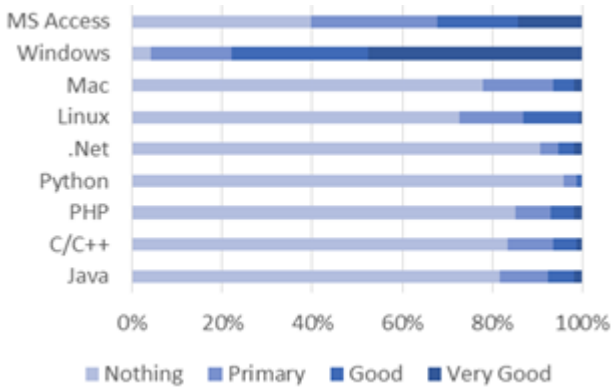


Figure 2. IT skill of computer science students prior Kankor.

4. CONCLUSION

Enrolment trends in Education and Higher Education generates vast amounts of data. With learning and tutoring management systems, the amount of data will be significantly increased either implicitly or explicitly. The main challenge preventing the applicability of EDM is lack of proper data storage and accessibility to data in electronic format. EMIS at MoE and HEMIS at MoHE together could be appointed to provide the raw data for EDM applications to help discern patterns of abilities and behaviors which could be used to help educational institutions.

5. ACKNOWLEDGMENTS

I thank my professors at Technical University of Berlin for their direct and indirect support, and the respondents.

6. REFERENCES

[1] Agnihotri Lalitha, Ott Alexander. 2012. Building a Student At-Risk Model: An End-to-End Perspective. In Proceedings of the 7th International Conference on Educational Data Mining, 209-212

[2] Andishman Mohammad Ikram. 2010. Modern Education in Afghanistan. Maiwand publication

[3] Central Statistics Organization. 2014-2015. Afghanistan Statistical Yearbook: Education Part One. Retrieved June 15, 2015 from http://cso.gov.af/en/page/1500/4722/2014-2015

[4] Emilio J. Castellano, Manuel J. Barranco, Luis Martínez. 2011. Academic Orientation Supported by Hybrid Intelligent Decision Support System, Efficient Decision Support Systems - Practice and Challenges from Current to Future.

[5] Pan Wei, Guo Shuqin, Alikonis Caroline, Bai Haiyan. 2008. Do Intervention Programs Assist Students to Succeed in College?: A Multilevel Longitudinal Study. College Student Journal 42, 1: 90-98

[6] Peroz Nazir, Tippmann Daniel. 2012. Information Technology for Higher Education in Afghanistan: ZiiK Report Nr. 32.

[7] Pratiwi Oktariani Nurul. 2013. Predicting student placement class using data mining. In Proceedings of 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering, 618–621.

Browsing-Pattern Mining from e-Book Logs with Non-negative Matrix Factorization

Atsushi Shimada
Kyushu University
Fukuoka, Japan
atsushi@artsci.kyushu-u.ac.jp

Fumiya Okubo
Kyushu University
Fukuoka, Japan
fokubo@artsci.kyushu-u.ac.jp

Hiroaki Ogata
Kyushu University
Fukuoka, Japan
ogata@artsci.kyushu-u.ac.jp

ABSTRACT

In this paper, we report our work-in-progress study about browsing-pattern mining from e-Book logs based on non-negative matrix factorization (NMF). We applied NMF to an observation matrix with 21-page browsing logs of 110 students, and discovered five kinds of browsing patterns.

Keywords

e-Book logs, pattern mining, non-negative matrix factorization

1. INTRODUCTION

An e-Book system can collect various kinds of operation logs when a page is opened, when the next page is browsed and so on. The analysis of e-Book logs enables teachers to understand how a student browses a given material. However, just giving or showing the logs is insufficient to understand behaviors of students because of their diversity and high dimensionality. In this paper, we apply non-negative matrix factorization (NMF) technique [2], which is known as akin to principal component analysis and factor analysis. In [1], NMF is utilized to extract a Q-matrix[1] from observed test outcome data for n question items and m respondents. In our study, we discover students' browsing patterns, i.e., how they browsed the given material, from e-book logs data for n page browse and m students. Besides, we analyze the relationship between the patterns and quiz scores.

2. E-BOOK LOGS

The e-Book logs were collected from 110 first-year students in an information science course taken in the first semester of the 2015 school year at Kyushu University in Fukuoka, Japan, via BookLooper (Kyocera Maruzen Systems Integration Co., Ltd.). Figure 1 shows samples of e-Book logs. There are many types of operations in logs, for example, OPEN means that the student opened the e-book file and

[1]A mapping of item to skills is termed a Q-matrix

User	Material	Operation	PageNo	Date	Time
X	00000000NLAT	OPEN	0	2014/10/15	9:01:09
X	00000000NLAT	CLOSE	1	2014/10/15	9:01:13
Y	00000000P82P	PREV	25	2014/10/29	10:05:35
Y	00000000P855	NEXT	2	2014/11/19	8:52:47
Z	00000000P84Z	NEXT	9	2014/11/12	9:31:30
...	...	...	...	...	...

Figure 1: Samples of e-Book logs

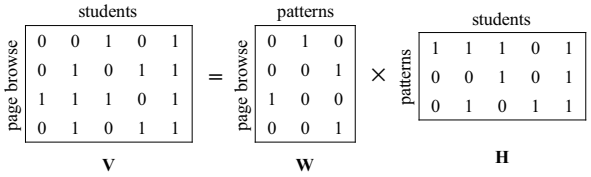


Figure 2: Pattern mining from browsing matrix

NEXT means that the student clicked the next button to move to the subsequent page. The duration of browsing each page can be calculated by subtracting the timestamps between subsequent pages.

3. METHODS

We utilize non-negative matrix factorization (NMF) technique to discover some browsing patterns. NMF approximately decomposes a matrix of $n \times m$ positive numbers V as the product of two matrices:

$$V \approx WH. \tag{1}$$

NMF imposes the constraint that the two matrices, W and H, be non-negative. In our approach, the matrix V, named browsing matrix, is represented by the fact whether a student browsed a page or not. More specifically, we set an element $v_{i,j}$ of the matrix V by

$$v_{i,j} = \begin{cases} 1 & (\text{if } t_{i,j} > th) \\ 0 & (\text{otherwise}), \end{cases} \tag{2}$$

where $t_{i,j}$ is the duration of page i browsed by student j. The decomposed matrices represent two latent relationships: "page browse vs. patterns" given by matrix W and "patterns vs, students" given by matrix H. In the sample of Figure 2,

Table 1: Description of discovered browsing patterns

pattern 1	browse the latter part of pages
pattern 2	browse the former part of pages
pattern 3	browse the middle part of pages
pattern 4	browse the beginning and end part of pages
pattern 5	browse pages between #12 and #15

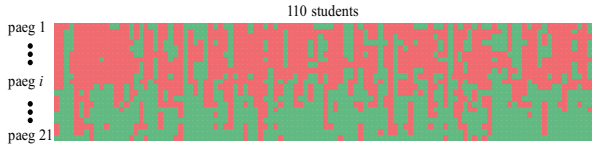


Figure 3: Browsing matrix. The red and green colors denote the value of $v_{i,j}$, red for one, green for zero, where the th was set to be 10 seconds.

browsing patterns are represented by three patterns in W. Meanwhile, H means whether a student has one or more browsing-patterns for a given material. In the experiments, we set the number of patterns to be five.

4. EXPERIMENTS

The browsing matrix used in our experiments were obtained from 110 first-year students. The students were asked to preview the material in advance before the lecture. They browsed the given material of information science which consists of 21 pages with a spread display setting. Therefore, the V is represented by 21-row $\times$ 110-column matrix as shown in Figure 3. The column of V corresponds to a student's previewing history whether he/she spent time at page i longer than th or not, which is calculated by formula (2). In the experiment, we set the th to be 10 (second).

NMF was performed to find five patterns. The decomposed matrices W and H are shown in Figure 4 and Figure 5 respectively. Note that the W is transposed in the figure due to the limitation of page space. Each pattern can be roughly described as Table 1.

The upper part of Figure 5 shows the correspondence between a student and his/her browsing pattern. The red color means that the student has the pattern (for example, the student in the most left column has pattern 2 and pattern 4). After the NMF, we acquired five groups based on consensus clustering technique (refer to literature [3] for more details). The bottom part of Figure 5 is the reordered matrix of W to show the group characteristics efficiently. The group 1 has the pattern 2 more strongly than the other patterns, which means that they spent longer time on the former part of pages.

We compared the student groups with their quiz scores (see Table 2). The quiz (max score = 10) was conducted at the beginning of the lecture. The average score of group 4 was lower than the other groups because the group had no pattern, i.e., they did not browse the material well. On the other hand, the group 2 got the highest score. They had the pattern 5, which corresponds to browsing the pages between #12 and #15. The contents of these pages were related

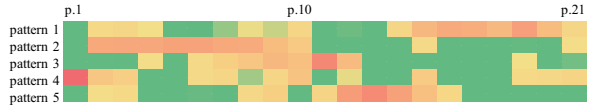


Figure 4: Visualized matrix W. Red parts represent the correspondence to each pattern. For example, pattern 2 denotes that pages from 2 to 10 are well browsed.

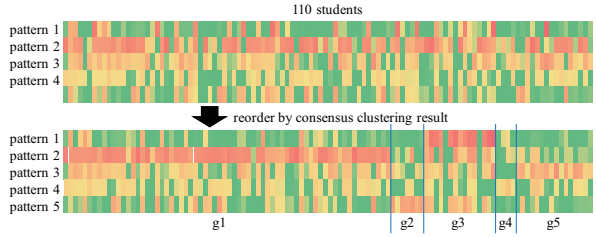


Figure 5: (Top:) Skill matrix visualized by color scale. The red color represents larger value. (Bottom:) Reordered pattern matrix based on consensus clustering result. There are five groups (g1, ..., g5) found by clustering.

Table 2: Average scores of quiz in each student group

student group	g1	g2	g3	g4	g5
average score	6.25	6.95	6.57	5.49	6.00

to the practice exercise to enrich the understanding. We guess that the students in group 2 could work the exercise because they had already understood the basic contents in the material. Therefore they got better quiz scores than the other student groups.

5. CONCLUSION

In this paper, we gave our work-in-progress report about e-Book browsing pattern mining and its potentials to fathom the relationships between patterns and understanding level of contents. In the experiments, we showed a primal result of pattern mining based on NMF. We found out that NMF could provide reasonable decomposed matrices to explain the browsing patterns. In the future work, we investigate the appropriate number of patterns because we predefined the number of patterns in this paper. Besides, we have to consider more effective method to generate a browsing matrix from e-Book logs.

6. REFERENCES

[1] M. Desmarais. Conditions for effectively deriving a q-matrix from data with non-negative matrix factorization. In Proceedings of the 4th International Conference on Educational Data Mining, pages 41–50, 2011.

[2] D. Lee and H. Seung. Learning of the parts of objects by non-negative matrix factorization. Nature, 401:788–791, 1999.

[3] S. Monti, P. Tamayo, J. Mesirov, and T. Golub. Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data. Mach. Learn., 52(1-2):91–118, July 2003.

How employment constrains participation in MOOCs?

Mina Shirvani Boroujeni, Łukasz Kidziński, Pierre Dillenbourg
Computer Human Interaction in Learning and Instruction
École polytechnique fédérale de Lausanne
{mina.shirvaniboroujeni, lukasz.kidzinski, pierre.dillenbourg}@epfl.ch

ABSTRACT

Massive Open Online Courses (MOOCs) changed the way continuous education is perceived. Employees willing to progress their careers can take high quality courses. Students can develop skills outside curriculum. Studies show that most of the MOOC users are pursuing or have received a university degree. Therefore it is beneficial to consider motives and constraints of this class of participants while designing a course. In this study we focus on time constraints experienced by full-time and part-time employees and students. Surprisingly, activities of students and employees are very similar regarding timing. We found that part-time employees spend more time on forum and are more active during the day. Employees are more active in the evening hours from Monday till Thursday. Based on our findings we suggest course design insights for practitioners.

1. INTRODUCTION

Time management in Massive Open Online Courses (MOOCs) is indispensable for success [2]. Recent studies show that difficulty with keeping up to deadlines is the main obstacle for engaging in a course [1]. Motivated by previous research, we assume that problems with time management are due to either professional constraints or issues with self-regulation [1] as illustrated in Figure 1. In this study we plan to provide a basis for understanding motives and limitations of MOOC participant depending on their employment status. Our general objective is to investigate: **How occupation (student, employee or part-time activity) influences participants time management in MOOC? How is it reflected by their engagement?**

2. DATASET

Our analysis is based on three successive offerings of an undergraduate engineering MOOC offered in Coursera entitled "*Functional Programming Principles in Scala*". The initial dataset contains 133,129 users. However information about the employment status is provided only by 8.7% of the participants. Based on this information, we extracted three categories of users: *full-time employed* (702 users), *full-time student* (110 users), *part-time activity* (66 users). 84% of full-employed participants hold a master or bachelor (45% and 39% respectively) and this ratio for the part-time group is 64% (32% and 32% respectively). Interestingly there is a noticeable percentage (22%) of participants with part-time activity who do not possess an academic degree.

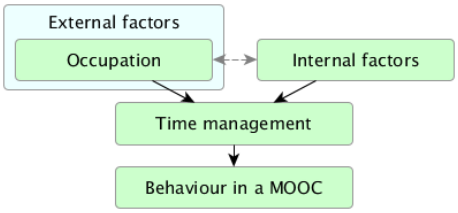


Figure 1: Time management is crucial for success in MOOC. We investigate the influence of occupation on time management.

For the analysis of users' performance we consider two types of events: watching videos and forum activities including viewing the forum (passive events) and writing or voting messages (active events). We extracted a set of features for each user, including final grade, count of forum events (total, active and passive), count of forum messages, average length of messages, count of submitted assignments and average number of attempts per assignment. In addition, we also extracted number of videos watched on different times of the day (Midnight, Morning, Midday, Afternoon, Evening, Night), different days of the week (Monday to Sunday) and different times of each week day. The final set includes 63 feature which were used in the analysis and building a predictive model in the following section.

3. FINDINGS

Q1. Are employed participants more likely to engage in the course? Based on χ^2 test, there is a significant relation between employment status and dropping out ($\chi^2 = 29.06, df = 2, p < 0.01$). According to the test residuals, among the three categories, employed participants are more likely to engage in the course, whereas students are most likely to drop out.

Q2. Do employed participants have higher achievement level? ANOVA on linear model of final grades re-

Proceedings of the 9th International Conference on Educational Data Mining

veals marginal significant difference between grades for students and employed participants (F[1, 810]=3.8, p=0.05): employed participants on average achieved a higher grade compared to the students (70 vs. 63 out of 100).

Q3. Are employed participants more engaged in forum? Total forum activity (active and passive events) by students and employed participants is similar, whereas part-time participants are significantly more active in forum compared to the other two groups (87 vs. 51, Mann-Whitney-Wilcoxon test, W=20516, p<0.01). Similarly number of posts by students and employed participants are not significantly different, while part-time participants have significantly more posts (M=4.6 vs. 1.7 posts, Mann-Whitney-Wilcoxon test, W=21282, p<0.01). Posts by part-time participants are the longest (M=83 words, t=-2.21, df=441.78, p= 0.02) and post by students are the shortest (M=53 words, t=3.14, df=239.35, p<0.01).

Q4. Do employed participant have different weekly pattern of activity? Distribution of videos watched on each week day shows that part-time participants watch more videos during the weekdays, whereas employed users and students are more active during weekends. Sundays and Mondays are the most active days for all groups and the activity level decreases from Monday to Saturday, mainly for employees and student. This trend could be related to the fact that video lectures were released on Sundays.

Q5. Do employed participants have different time distribution of activities? Number of videos watched in different parts of the day shows to be related to the employment status of participants ($\chi^2 = 109, df = 10, p < 2.2e-16$). As shown to Figure 2, employed participants are the most active group during evening hours (F[1, 876]=4.92, p=0.02), students are the most active group during night hours and part-time participants are the most active group during mid-day. Furthermore unlike part-time participants, the activity level of the other two groups is higher during the afternoon and evening compared to the mid-day hours.

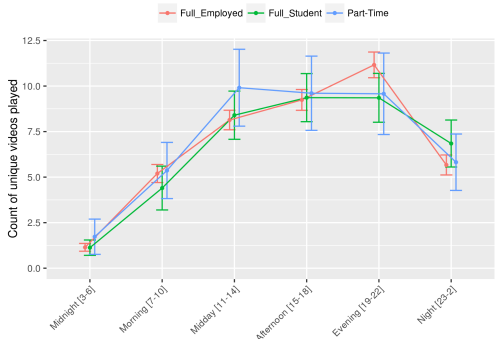


Figure 2: Distribution of number of videos watched at different times of the day.

Further investigations of participants' activity patterns in different days of the week reveals that the observed evening activity peak for the employed participants is related to the working days (Monday to Thursday). On Friday their overall activity level is low and on weekends their activity peak time is shifted to the afternoon hours. Remarkably, all groups are active in the mornings and during the midday. In particular, this could suggest that full-time employees engage in MOOCs during the morning commutes and also during the work day. Nevertheless this finding should be further confirmed in interviews with MOOC participants.

Q6. To what extent can we predict user's employment status based on derived features? In order to predict employment status of participants based on the features described in Section 2, we trained several classifiers including Neural Network, Penalized Multinomial Regression, Random Fores and Support Vector Machine with linear kernel. Using 10-fold cross validation, the highest Cohen's κ (0.45) was achieved by Random Forest classifier.

4. CONCLUSION

Our analysis revealed that employment is reflected by different activity patterns. This confirms our hypothesis that time constraints influence user's participation in MOOCs. Our findings partially confirm previous theories. In particular, higher drop-out rate from MOOCs among students versus employees can be attributed to lower academic and social commitment [3]. This phenomenon can also be linked to better time management of employees (participation in MOOC during the evening just after work) [2]. Further controlled studies should be conducted to discover true causality.

Based on the insight from our analysis, we suggest following design considerations while designing MOOCs courses: **(1) Choose the lecture release day depending on the target audience.** We found that activity of employed participants drops during the weekdays. On the other hand, video release on Sunday make participants work on Monday despite the general lower activity during workdays. Therefore, releasing lectures on Saturday might increase overall activity. **(2) Choose activities convenient for commute time and short sessions.** Our analysis showed activities during potential commuting hours, therefore designing short and mobile-friendly videos and activities could facilitate users engagement during this time. **(3) Choose accurate timing for communication with users,** such as the time when they are most likely to visit the MOOC **(4) Include temporal activity indicators in predictive models,** as time-related features showed to be correlated not only with employment status but also with the success in a MOOC.

5. REFERENCES

[1] René F Kizilcec and Sherif Halawa. Attrition and achievement gaps in online learning. In *Proceedings of the Second (2015) ACM Conference on Learning@ Scale*, pages 57–66. ACM, 2015.

[2] Ilona Nawrot and Antoine Doucet. Building engagement for mooc students: introducing support for time management on online learning platforms. In *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pages 1077–1082. International World Wide Web Conferences Steering Committee, 2014.

[3] Vincent Tinto. Dropout from higher education: A theoretical synthesis of recent research. *Review of educational research*, 45(1):89–125, 1975.

Quantifying How Students Use an Online Learning System: A Focus on Transitions and Performance

Erica L. Snow
Center for Technology in Learning
SRI International
333 Ravenswood Ave
Menlo Park ,CA
erica.snow@sri.com

Andrew E. Krumm
Center for Technology in Learning
SRI International
333 Ravenswood Ave
Menlo Park ,CA
andrew.krmm@sri.com

Timothy Podkul
Center for Technology in Learning
SRI International
333 Ravenswood Ave
Menlo Park, CA
timothy.podkul@sri.com

Mingyu Feng
Center for Technology in Learning
SRI International
333 Ravenswood Ave
Menlo Park, CA
mingyu.feng@sri.com

Alex J. Bowers
Teachers College
Columbia University
525 West 120th St.
New York, NY 10027
bowers@tc.edu

ABSTRACT
The current study employs transitional probabilities as a way to classify and trace students' interactions within an online learning system. Results revealed that students' interaction patterns within the system varied in relation to their performances on embedded assessments. The results and methodologies presented here are designed to provide practitioners with a starting place for how to extract information concerning how and why their students interact within an online environment.

Keywords
Blended Learning, Transitional Probabilities, Online Technology

1. INTRODUCTION
The use of blended learning techniques has become increasingly prevalent within high school classrooms [1]. One goal of blended learning is that information concerning students' behaviors and performance within various technologies can be used to inform instructional practice [2]. However, trace-level data from most technologies are often inaccessible or unusable for practitioners [3]. The current work aims to better understand what methodologies and tools are useful for helping practitioners make sense of *how* students interact with assessments and resources within online technologies. Using transitional probabilities we examined how 812 middle and high school students interacted with an online learning system (OLS) as part of their regular Math classroom instruction and how these behaviors varied as a function of students' performance within the system.

2. METHODS

2.1 Participants
The participants included 812 students from a large charter management organization (CMO) in the San Francisco Bay area. Over 60% of students who attend this CMO come from underserved populations (e.g., African American and Hispanic or Latino) and over 40% qualify for free or reduced priced lunches. The participating students regularly interact with the OLS as part of their Math curriculum.

2.2 Procedure, Measures, and Data Processing
Students interacted with the Math content on the OLS throughout the 2014-2015 school year. In the work presented here we examined how students interacted in one lesson for their Math curriculum, *Linear Equations*. During this lesson, students could freely choose to engage in a variety of activities at their own pace. These activities can be grouped into three categories that represent a different type of functionality within the system; these functionalities are *Post Assessments* (Linear Equation content gleaned from system resources), *Pre Assessments* (baseline measure of students' Linear Equation knowledge), and *Resources* (unique items –PDFs, videos, images- that provide Linear Equation content). These categories afforded the opportunity to trace students' choice of interactions within the system while also providing a means of surfacing reoccurring patterns of behavior that students exhibit throughout the school year. All interactions are logged within the system and provide valuable insight into *how* students interact with the OLS.

3. QUANTITATIVE METHODS
To examine variations in students' behavior patterns within the Linear Equation curriculum of the OLS, transitional probabilities were conducted. This analytical tool provides a means to provide teachers with a visualization of students' learning trajectories. This is particularly useful for practitioners interested in examining how closely students' choices followed the intended system curriculum. The following section provides a brief description and explanation of transitional probabilities and their application to the current data set.

3.1 Transitional Probabilities
Transitional probabilities were calculated using a statistical sequencing procedure established in D'Mello, Taylor, and Graesser (2007; [4]). This sequencing procedure is calculated using the formula $L[I_t \rightarrow X_{t+1}]$. In this formula, L is the likelihood function of the student's current choice in the system (I) at specific time point t, and X is their next interaction choice at the next time point (t+1). Thus, this sequencing procedure surfaces the probability of a student's interaction choice given their previous choice. For instance, if Zach chooses to take a Pre

Assessment, the above formula will be used to surface what choice Zach is most likely to choose next (e.g., another Pre Assessment, a Post Assessment, or a Resource). These probabilities were calculated for each of the 812 students, which resulted in a unique pattern of choices for each student. The results reported below address students' interactions with the Pre Assessment, Post Assessment, and Resources associated with Linear Equations content within a 9th grade Math course.

4. RESULTS

Overall, 812 students interacted with the Linear Equation content within the OLS system. Teachers recommended that students take the Pre Assessment, interact with system Resources, and then take the Post Assessment to measure changes in learned material. However, as this was a blended learning environment students were free to choose how they would spend their time and what features they would interact with. Using system log data, we classified students' interactions into one of three orthogonal categories (i.e., Post Assessments, Pre Assessments, and Resources). We classified students as passing if they scored at or above 80% and failing if the scored below 80%. To examine how students interacted with the system, we calculated the total frequency of students' interactions with each of these three categories. On average, students made 38 interactions within the system and spent the majority of their time interacting with Pre Assessments (53%), followed by taking Post Assessments (32%) and interacting with Resources (15%).

4.1 Interaction Transitions

The current work aimed to better understand how students' performance in Math 9 influenced their next interaction within the OLS. Figure 1, displays the conditional transition probabilities for students who passed a Post Assessment for Linear Equations. In this figure, there are three possible interactions, retrying a Post Assessment, transitioning to a Pre Assessment, or transitioning to a Resource. Students can also choose to move onto another topic. This analysis revealed that after students' passed a Post Assessment, .01% of the time they tried another Post Assessment, 1% of the time they took a Pre Assessment, and 17% of the time they interacted with a Resource. Most often after passing a Post Assessment, students left that content area to start another (72%).

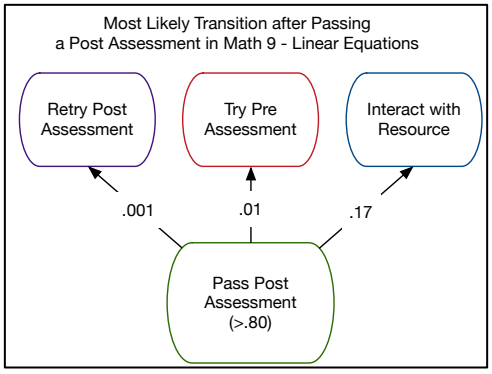


Figure 1. Conditional probabilities after passing Post Assessment.

Figure 2, displays the conditional transition probabilities if a student fails a Post Assessment for Linear Equations. Similar to Figure 2, there are three possible interactions along with students' choice to leave the curriculum. This analysis revealed that after students' failed a Post Assessment, 48% of the time they retook the Post Assessment, 43% of the time they took a Pre Assessment and 7% of the time they interacted with a Resource. Unlike students who passed a Post assessment (Figure 1), students who failed a Post Assessment were less likely to exit the curriculum (2%) and instead most often interacted with another form of assessment (Pre or Post).

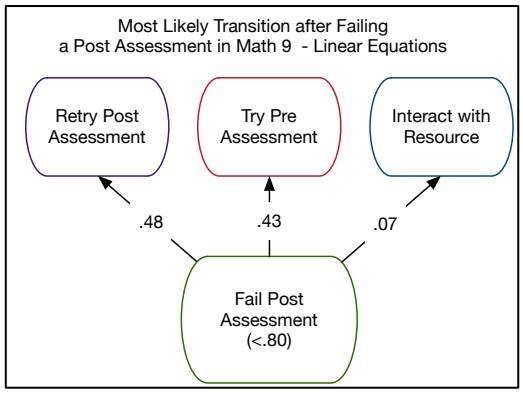


Figure 2. Conditional probabilities after failing a Post Assessment.

5. DISCUSSION

These exploratory findings are promising for both educational researchers and practitioners as they reveal how students' behavior patterns manifest and vary as a function of performance. The current work begins to shed light upon the nuanced ways in which students' interactions can be traced and classified within online environments. In the future, this work will be expanded to examine students' behavior patterns across multiple classrooms and courses. The goal will then be to examine how students' behaviors vary as a function of performance and domain. This information may prove useful to practitioners wishing to better understand how information extracted from technology can be used to inform instructional practices.

6. ACKNOWLEDGEMENTS

This research was supported by the National Science Foundation (DRL-1444621). The opinions expressed are those of the authors and do not necessarily represent views of the NSF.

7. REFERENCES

[1]. Stockwell, B. R., Stockwell, M. S., Cennamo, M., & Jiang, E. 2015. Blended learning improves science education. Cell, 162(5), 933-936.

[2]. Halverson, R., Grigg, J., Prichett, R., & Thomas, C. 2007. The new instructional leadership: Creating data-driven instructional systems in school. Journal of School Leadership, 17(2), 159.

[3]. Jacovina, M. E., Snow, E. L., Allen, L. K., Roscoe, R. D., Weston, J. L., Dai, J., & McNamara, D. S. 2015. How to visualize success: Presenting complex data in a writing strategy tutor. In O. C. Santos, J. G. Boticario, C. Romero, M. Pechenizkiy, A. Merceron, P. Mitros, J. M. Luna, C. Mihaescu, P. Moreno, A. Hershkovitz, S. Ventura, & M. Desmarais (Eds.), Proceedings of the 8th International Conference on Educational Data Mining (Madrid. Spain) EDM 2015 pp. 594-595.

[4]. D'Mello, S. K., Taylor, R., and Graesser, A. C. 2007. Monitoring affective trajectories during complex learning. In D. S. McNamara & J. G. Trafton (Eds.), Proceedings of the 29th Annual Meeting of the Cognitive Science Society (Nashville, Tennessee, August 1-4, 2007) Cognitive Science Society, 203-208.

A Platform for Integrating and Analyzing Data to Evaluate the Impacts of Educational Technologies

Daniel S. Stanhope
Lea(R)n, Inc.
Raleigh, NC 27603
daniel.stanhope@learntrials.com

Karl T. Rectanus
Lea(R)n, Inc.
Raleigh, NC 27603
karl@learntrials.com

ABSTRACT
Educational technology (edtech) products are ubiquitous in schools, but a paucity of research has evaluated their impact on education outcomes. Herein we describe a platform (i.e., LearnPlatform) that enables users to integrate and analyze data to rigorously evaluate the impacts of edtech. The platform also enables users to mine large and diverse datasets to identify patterns and trends in edtech usage and impact, and to build statistical models through predictive analytics that use multiple predictors to forecast future events, trends, and probabilities. Ultimately, educators and researchers can use LearnPlatform to generate evidence-based insights about edtech ecosystems within and across schools, districts, and states, which will improve the discovery, purchasing, and evaluation of edtech products in myriad educational contexts.

Keywords
Educational technology, efficacy, data, evaluation, education outcomes

1. INTRODUCTION
Educational technology (edtech) is increasingly pervasive. Each year, billions of dollars are spent and innumerous products are released. Despite immense resources invested, there has not been a standard system for monitoring and evaluating the use, quality, and efficacy of edtech products, leaving school leaders without access to critical data when making instructional, operational, and fiscal decisions. These decision makers need timely, reliable, evidence-based information on edtech interventions to know what to buy, how to support instruction and implementation, and how to improve student outcomes. Accordingly, Lea(R)n, Inc. worked with thousands of educators, state and district leaders, subject matter experts, and researchers to develop an online edtech management platform, called LearnPlatform, to help education organizations and institutions understand and manage which edtech products are best for their needs.

2. EDTECH MANAGEMENT PLATFORM
LearnPlatform is an edtech management platform that helps schools and districts understand which edtech products are best for their classrooms and students. To ensure valuable and trustworthy insights, the platform was built to support sound research methods and study designs[1] that enable systematic investigations within authentic educational contexts. The platform offers a research-based system for educators to understand, manage, and evaluate edtech products. Among other things, the platform allows users to (a) identify, catalogue, and monitor the products that are being used in their classrooms; (b) grade products on a valid and reliable rubric;[2] (c) connect with colleagues to share insights and ask questions; and, (d) conduct edtech evaluations that range from rapid-cycle pilots to randomized control studies (RCTs) to multi-product factorial studies. The analytics module of the platform, called LearnTrials IMPACT (*Integrating Metrics for Producing Analytics on Classroom Technology*), allows users to rapidly integrate disparate datasets and analyze those data to generate evidence-based insights on edtech interventions.

3. ANALYTICS MODULE
The platform's analytics module (LearnTrials IMPACT) has several noteworthy components. First, the platform maintains and continuously updates a relational database with over 4,000 edtech products that are available to educators (see Figure 1).[3]

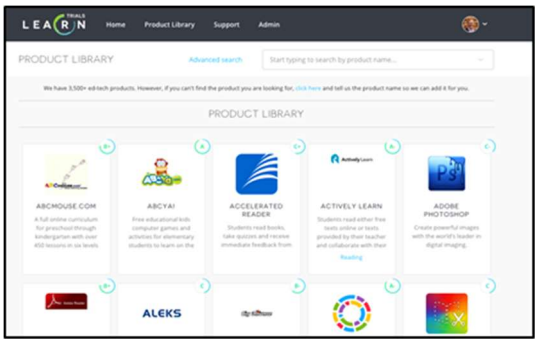


Figure 1. Screenshot of product library with product grades.

Second, a structured architecture allows educators to leverage useful features, including managing portfolios of products, sharing experiences with tools, asking colleagues questions, viewing products' grade reports, and comparing products side by side (see Figure 2 for example of an administrator view).

Third, capabilities of the platform allow districts to collect rapid feedback on the products they already use, launch evaluations of products, and analyze findings filtered by dozens of criteria (e.g., purpose of product use, frequency of use, student groups with which the product is used; see Figure 3).

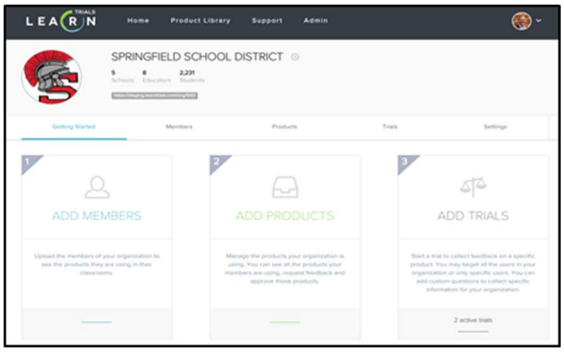


Figure 2. Administrator view of LearnPlatform.

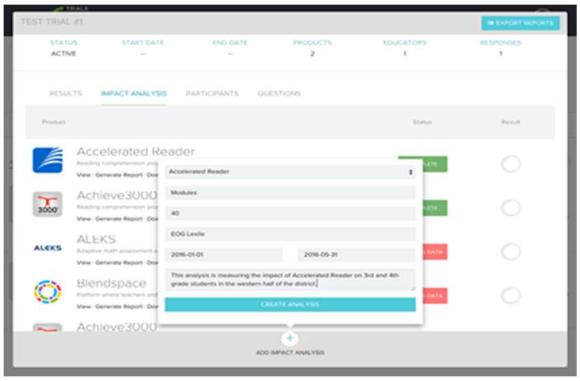


Figure 3. Screenshot of functionality in the IMPACT layer.

Fourth, the platform aggregates educators' evaluations of products into interpretable and actionable recommendations about the product and its optimal use with various student populations. Finally, a data integration and automated analytics layer allows users to rapidly de-identify, upload, and analyze product usage (e.g., time on system, modules completed), student outcomes (e.g., achievement, motivation, engagement), and other data to produce dynamic reports and dashboards that inform instructional, operational, and budgetary decisions (see Figure 4 for example of Impact Analysis Report with simulated data and a fake product).

4. CASE STUDY

Schools, districts, and states across the US are using LearnPlatform. One of the nation's largest school districts leveraged LearnPlatform to conduct a controlled trial with a quasi-experimental design that generated insights for budgeting and implementation. In the efficacy trial, the district studied a widely used edtech product for elementary literacy. The sample included 18 schools who used the product (treatment group; $n_T > 8{,}000$) and 18 schools who did not use the product (control group; $n_C > 8000$). We tested for baseline equivalence on multiple measures, including demographics and prior achievement. We also applied statistical adjustments to control for variance attributable to extraneous factors and covariates. We first computed covariate-adjusted effect sizes to determine the extent to which the product exhibited an impact on the treatment versus the control, then conducted cluster analysis to identify student clusters of product usage and examined achievement for different clusters. Results were confirmed through a separate, blind analysis by the district's data and accountability office. Additional analysis of costs informed the district's purchasing and budgeting decisions.

Figure 4. Example of an impact report (fake product and school).

5. FUTURE DIRECTIONS

First, LearnPlatform will enable users to mine datasets to identify patterns and trends in edtech usage and impact, and to build statistical models through predictive analytics that forecast future events, trends, and probabilities. Second, once enough data are available, users will be able to leverage LearnPlatform to conduct meta-analyses to begin to elucidate conditional and contextual effects that may differentiate the efficacy of a given intervention based on factors that vary across schools, districts, or states. Ultimately, educators and researchers will use LearnPlatform to gain data-driven insights into edtech ecosystems across schools, districts, and states, and to improve discovery, purchasing, and evaluation of what works for educators and their organizations.

6. ACKNOWLEDGMENTS

Development of the IMPACT layer has received funding from organizations such as the Bill and Melinda Gates Foundation.

7. REFERENCES

[1] Lea(R)n, Inc. (2015, November 8). Grading EdTech: Our Rubric Effectively Differentiates Products. Retrieved from http://go.learntrials.com/rubric-research/

[2] Singer, N. (2016, January 17). Education Technology Graduates From the Classroom to the Boardroom. Retrieved from http://www.nytimes.com/2016/01/18/technology/education-technology-graduates-from-the-classroom-to-the-boardroom.html?_r=1

[3] Creswell, J. W. (2013). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches (4th ed.)*. Thousand Oaks, CA: SAGE Publications

Patterns of Usage from Educational Technology Products across America

Daniel S. Stanhope
Lea(R)n, Inc.
310 S. Harrington St.
Raleigh, NC 27603
daniel.stanhope@learntrials.com

Karl T. Rectanus
Lea(R)n, Inc.
310 S. Harrington St.
Raleigh, NC 27603
karl@learntrials.com

ABSTRACT

Educational technology (edtech) products are ubiquitous in schools, yet there is a dearth of research examining their use and efficacy. This leaves schools and districts without evidence to inform important decisions about edtech budgeting, instruction, impact, and implementation. We report results from a study that uncovered startling trends in edtech usage across multiple paid products and dozens of schools. Notably, 36.6% of purchased student licenses were never used. An additional 28.2% of the licenses were used negligibly, failing to meet a quarter of the fidelity goal set by the product companies or districts. Further, anecdotal evidence suggests school- and district-level leaders are unaware of these realities. This suggests a vast amount of resources are being unknowingly squandered or misallocated. Combined with analysis of how product usage impacts student achievement, these results demonstrate how schools and districts can utilize data to understand and manage their edtech ecosystems while improving critical edtech decisions.

Keywords

Educational technology, efficacy, fidelity, evaluation, education

1. INTRODUCTION

Educational technology (edtech) presents both opportunities and challenges for educators and their organizations. Challenges include allocating resources appropriately, implementing products with fidelity, and ensuring product efficacy. Unfortunately, these challenges have been exacerbated because heretofore districts have not had systems or methods for collecting, comparing, and analyzing disparate data sources in a way that informs budgetary or instructional decisions. To address that lack of evidence, schools and districts across the nation have been using LearnTrials—a module on the LearnPlatform—to measure an integrated system of data and variables, enabling them to generate key insights and rapidly make informed decisions. In this paper, we report a specific set of early findings from a synthesis of systematic research focusing on edtech usage patterns, and we discuss the implications for implementation, impact, and budgeting.

More than $8 billion (PreK-12 alone) are spent annually on edtech products in the US with the goal to improve important education outcomes.[1] Both producers and consumers of edtech products worry about using them with fidelity—that is, ensuring students receive the "recommended dosage" to achieve the intended outcomes. Most agree that implementation and its impacts on budget and achievement are interrelated and worthy of treatment as a system; however, limited research has examined fidelity of edtech usage. This has led dozens of schools and districts to use LearnTrials to conduct rapid, cost-effective evaluation of multiple products, analyzing both edtech usage and efficacy.

2. METHODS

2.1 SAMPLE

The sample for this study is 49 K-12 schools in multiple districts and states. Overall, the sample included over 17,000 students from a diverse set of schools. For each school, we examined data on product usage collected during the 2014-2015 academic year. Specifically, we tracked the extent to which students used their licenses for six well-known digital math and literacy tools. Each of these products was well-established in the marketplace, used for primary instruction (rather than supplemental), and ranged in price from $16 to over $100 per student, per year.

2.2 ANALYSIS

The main analysis for this study involved descriptive statistics on the extent to which students used their product licenses. Each of the six products prescribe a specific amount of student usage, often called the recommended dosage. In other words, these products have predetermined metrics for usage goals (e.g., time logged in, progress through syllabus, number of lessons passed) intended to promote marketed outcomes. Based on these measures, we analyzed the extent to which students met certain expectations. Specifically, we examined whether students (a) never used the product, (b) used the product but failed to meet even 25% of the goal, (c) met 25% of the usage goal, (d) met 50% of the usage goal, or (e) fully met the usage goal.

3. RESULTS

We found consistent patterns of usage across the schools and across the products. The main finding: 36.6% of purchased product licenses were never activated. An additional 28.2% of students activated their license, but did not use the product enough to meet even 25% of the established goal. Thus, approximately 64.8% of students exhibited zero or trivial use. Moreover, only 5.2% of students actually received the full recommended dosage (Figure 1; see Figure 2 for a breakdown of

use by product). In summary, schools are paying significant amounts of money for products that students are not using.

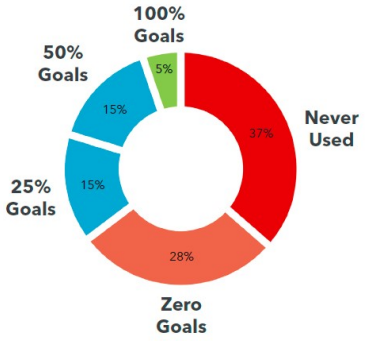


Figure 1. Percent of paid product licenses meeting dosage goals.

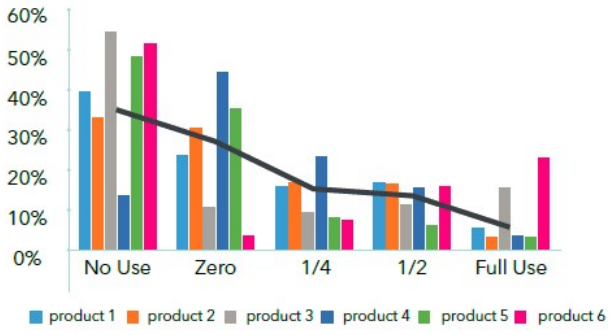


Figure 2. Paid product licenses meeting dosage goals by product. (Product names undisclosed for sake of anonymity.)

4. DISCUSSION

To be clear, the startling lack of product usage across schools is not an indictment of edtech products or the schools that use them—classroom technologies are valuable, and have the potential to amplify learning. While these are early findings, they have numerous implications for schools and districts.

Implementing learning technologies in schools and districts presents opportunities and challenges. One way to maximize the former and minimize the latter is understanding important contextual factors. Recognizing the specific factors that impact use within local contexts can uncover opportunities for growth. Structured pilots, rapid feedback cycles, and scaled roll-outs do not have to be cumbersome. Leveraging data-rich product pilots can address common challenges. By using research-backed, standardized edtech management systems in their local contexts, districts can lower opportunity costs, reduce negative impacts on teaching and learning, and mitigate political consequences of "all-in, all-at-once" implementations.

Understanding product efficacy—the extent to which a product impacts intended educational outcomes—is important. The U.S. Dept. of Education, the Bill and Melinda Gates Foundation, and others have recently invested in rigorous and realistic evaluation of products at every stage. If students do not use a product, they cannot capitalize on its potential benefits. Discovering that edtech products are consistently underused (or never used) is a first step. Providing schools and districts insights into situational variables (e.g., student characteristics, school types, demographics, or pedagogical styles) would help educators and product companies understand the contexts in which products have positive, negative, or negligible impact. Our research has shown times when minimal (and even significant) usage had deleterious effects on student achievement. In other cases, specific student groups using certain edtech products saw greater gains than did their peers. Delivering context-specific insights that are based on statistical analysis via timely, easy-to-understand dashboards and reports help schools and districts identify the best tools for their situations and instructional needs.

A final implication is the obvious impact on budget. If we extrapolate the findings reported herein, it is likely that last year schools spent nearly $3 billion on product licenses that were never activated (37% of the $8 billion spent across U.S. schools). However, edtech purchasing decisions do not exist in a vacuum; rather, they are richly contextualized and made based on budgetary constraints, merit of competing products, politics, and precedent. Challenges also include current business models, lack of pricing transparency, and unknown usage data. Furthermore, edtech purchasing has decentralized rapidly, meaning individual educators and schools are making more decisions, which creates organizational challenges for district and state leaders.

Educators and their organizations need a systematic approach for gathering evidence,[2] and for rapidly understanding organization-wide product usage and efficacy. Analysis of local data as well as analysis of large-scale databases can greatly enhance our ability to evaluate edtech phenomena.[3] Then, implementing edtech management systems, service level agreements, and performance contracts (based on successful usage or other measurable milestones) are not only possible, but also capable of improving instruction, finances, and educational outcomes.

The consistent patterns of usage—specifically the limited use of paid licenses—across edtech products in education environments offers a massive opportunity to improve a complex system. Until recently, edtech decisions lacked a systematic approach for measuring and collecting evidence on the most important variables. However, dozens of schools and districts are using the edtech management LearnPlatform and its LearnTrials module to analyze their edtech ecosystems in unbiased and rapid ways, so they can make evidence-based decisions that enhance the fidelity of implementation, boost product impact on student achievement, and maximize resources (e.g., time and money).[4]

5. REFERENCES

[1] Richards, J. & Stebbins, L. (2014). 2014 U.S. Education Technology Industry Market: PreK-12. Washington, D.C.: Software & Information Industry Association.

[2] Coburn, C. E., Honig, M. I., & Stein, M. K. (2009). What's the evidence on districts' use of evidence? In J. D. Bransford, D. J. Stipek, N. J. Vye, L. M. Gomez, & D. Lam (Eds.), The role of research in educational improvement (pp. 67-87). Cambridge, MA: Harvard Education Press.

[3] Penuel, W. R., & Means, B. (2011). Using large-scale databases in evaluation: Advances, opportunities, and challenges. *American Journal of Evaluation*, 32, 118-133.

[4] Johnson, K. (2016, March 15). Resources to Help You Choose the Digital Tools Your Classroom Needs. Retrieved from https://www.edsurge.com/news/2016-03-15-resources-to-help-you-choose-the-digital-tools-your-classroom-needs

Learning curves versus problem difficulty: an analysis of the Knowledge Component picture for a given context

Brett van de Sande
Pearson Education
brett.vandesande@pearson.com

ABSTRACT

The Knowledge Component (KC) picture of learning has proven useful for constructing models of student learning in a number of subject areas. However, it is still unclear how well this picture generalizes to other contexts and subject areas. A corpus of 62,000 exercises for 10 textbooks on the Mastering platform has been tagged by content experts. In this report, I introduce a strategy for investigating the importance of a given set of KCs in describing student performance as the students solve problems. The strategy is to see how much of the student's performance on an exercise is explained by the associated KC and how much it is predicted by a problem-specific difficulty parameter. To do this, I introduce a model that is a combination of the Rasch model and the learning curves from the KC picture. For this corpus and set of KC tags, a rather striking picture emerges: problem difficulty accounts for most of the student behavior while KC learning accounts for only a small portion of the student behavior. I hypothesize that these KC tags do not accurately capture the skills students are using while doing their homework.

Author Keywords

Learning Curves, Knowledge Components

ACM Classification Keywords

I.2.6 Learning: Knowledge acquisition

Knowledge components (KCs) are bits of information needed to solve a problem [5, 2]. KCs generally have some sort of pre-requisite relations. However, aside from prerequisites, a KC can, by definition, be mastered independently from other KCs. This definition assumes that KCs are *context independent*. That is, the student's ability to apply that KC correctly or quickly does not depend on the particular problem the student is solving or the other KCs needed to solve that problem.

Since KCs are *defined* to have these properties, then it remains to be seen whether a given set of KC labels for a particular curriculum provides a useful description of skill acquisition. Much of the pioneering work on KCs focused on middle school math [4]. It is unclear whether this picture extends to the corpus examined here.

Table 1. Some Knowledge Components for Chapter 32 of "University Physics" by Young, Freedman, and Lewis [6].

1	Relationship between speed of electromagnetic (EM) waves, wavelength and frequency
2	Writing Maxwell's equations for free space. Using Faraday's Law.
3	Direction of propagation of an electromagnetic wave

One way to determine how well the KC picture is working is to examine the associated learning curves. If the curves increase/decrease more-or-less monotonically (depending on the measure of competence) then the KC picture is working. A smooth learning curve implies that the associated KCs account for most of the student performance on a problem while other aspects of the problem are less important.

A corpus of over 62,000 exercises on the Mastering platform has been tagged by content experts. This corpus covers homework exercises for 10 college-level textbooks in anatomy and physiology, biology, organic chemistry, general chemistry, and physics. An typical set of KCs is shown in Table 1. On average, there are about a dozen KCs per chapter.

We examined log data from problems solved on the Mastering platform during the Spring of 2014. We selected students whose coursework spanned more than 25 days and who were enrolled in a course containing more than 50 students.

Before we address the main question of the validity of the KC picture for this corpus, we mention some general properties of the log data. The learning curves (see Fig. 1) are expressed in terms of "difficulty" which is defined to be minus the logistic of the probability of "correct on first try."

The mean number of opportunities to practice a given KC is 3.84, averaged over students and KCs. So, students have very few opportunities to practice a given KC.

Also, the number of students practicing a KC usually decreases rapidly with increasing opportunity number t. This can result in a selection bias, since the population is changing with t. Thus, to produce a learning curve for a given KC, we rank the students by the total number of opportunities for that KC and take the uppermost portion as our student population. An example learning curve is shown in Fig. 1. In general, we

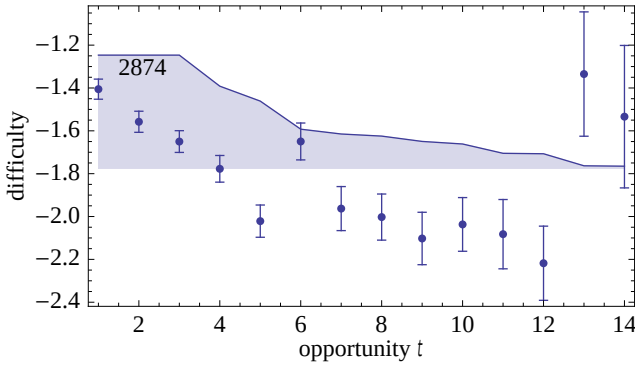


Figure 1. Learning curves for the first KC listed in Table 1. Difficulty should decrease as students learn. The shaded region represents the relative number of students who completed that opportunity and the number in the upper left corner is the initial number of students.

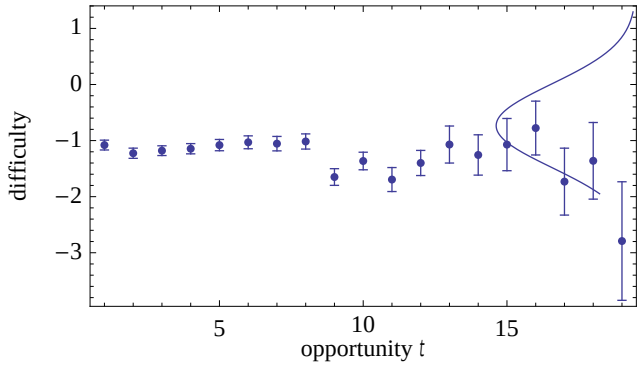


Figure 2. KC difficulties $\zeta_{k,t}$ versus opportunity number t from the Rasch/KC model applied to student log data for the first KC in Table 1. The curve on the right is a gaussian that represents the distribution of problem difficulties for the exercises labeled with the associated KC.

find that learning curves are not monotonically decreasing. In fact most do not even show a decreasing trend.

There must be important aspects of the exercises that are not captured by these KCs. Thus, we introduce problem difficulty β_p to capture the aspects of a problem not explained by the KCs. This leads us to introduce the Rasch/KC model: a hybrid of the Rasch model [3], and the learning curve picture.

If $P_{s,p}$ is the probability that student s gets problem p correct, then we define $P_{s,p}$ by the logistic equation:

$$\text{logit}\left(P_{s,p}\right) = \theta_s - \beta_p - \sum_{(k,t)\in\mathcal{T}_{s,p}} \zeta_{k,t} \qquad (1)$$

where θ_s is the skill of student s, β_p is the difficulty of exercise p, and $\zeta_{k,t}$ is the difficulty of applying KC k on opportunity t. $\mathcal{T}_{s,p}$ is the set of KC, opportunity pairs where $(k,t) \in \mathcal{T}_{s,p}$ means that problem p is opportunity t for student s to apply KC k. The log-likelihood for a set of students and problems to obtain a particular set of outcomes is

$$\log\left(\mathcal{L}\right) = \sum_{s,p\in\mathcal{C}_s} \log\left(P_{s,p}\right) + \sum_{s,p\in\mathcal{I}_s} \log\left(1 - P_{s,p}\right) + \quad (2)$$

where $\mathcal{C}_s$/$\mathcal{I}_s$ is the set of problems s got correct/incorrect.

If we drop $\zeta_{k,t}$, then we obtain the usual Rasch model. Likewise, if we drop θ_s and β_p and fit the resulting model to student data, a plot of $\zeta_{k,t}$ versus opportunity t will yield the conventional learning curve for KC k; this is precisely what we have plotted in Fig. 1. This model is similar to the Additive Factors Models (AFM) [1] except that AFM restricts $\zeta_{k,t}$ to be linear in t.

We can apply this model to student log data associated with the KCs listed in Table 1. We find that both student skills $\{\theta_s\}$ and problem difficulties $\{\beta_p\}$ are Gaussian distributed with standard deviations of 1.02 and 1.15, respectively.

Looking at the KC difficulties $\zeta_{k,t}$ in Fig. 2 we see that the difficulties vary little with opportunity number. We also, see that the associated problem difficulties, represented by the Gaussian distribution on the right, vary significantly more than the KC difficulties. The same qualitative behavior is seen for all KCs we have analyzed. We conclude that, for this corpus and KC labeling, problem difficulty is much more important than KC mastery when predicting student performance on an exercise.

If we look at the KCs, see Table 1, we see that they represent content knowledge rather than more abstract problem solving skills. It may be that the students have already learned the content knowledge in lecture or reading and, during their homework, they are really learning how to apply that content knowledge to various physical situations. If this is the case, it may be more appropriate to label problems with labels that are more oriented towards problem-solving skills, like "given description of situation, determine that one should relate velocity, frequency, and wavelength." Also, it may mean that one can explain student performance with just a few KCs like "solve physics word problem" or "solve problem with kinematics graphs."

REFERENCES

1. Chi, M., Koedinger, K., Gordon, G., Jordan, P., and VanLehn, K. Instructional Factors Analysis: A Cognitive Model For Multiple Instructional Interventions. In *Proceedings of the 4th International Conference on Educational Data Mining* (Eindhoven, the Netherlands, June 2011).
2. Koedinger, K. R., Corbett, A. T., and Perfetti, C. The Knowledge-Learning-Instruction Framework: Bridging the Science-Practice Chasm to Enhance Robust Student Learning. *Cognitive Sci. 36*, 5 (2012), 757–798.
3. Rasch, G. *Probabilistic Models for Some Intelligence and Attainment Tests.* MESA Press, Chicago, 1993.
4. Ritter, S., Anderson, J. R., Koedinger, K. R., and Corbett, A. Cognitive Tutor: Applied research in mathematics education. *Psychon. B. Rev. 14*, 2 (Apr. 2007), 249–255.
5. VanLehn, K. The Behavior of Tutoring Systems. *Int. J. Artif. Intell. Ed. 16*, 3 (Jan. 2006), 227–265.
6. Young, H. D., Freedman, R. A., and Ford, A. L. *University Physics with Modern Physics*, 13 edition ed. Addison-Wesley, Boston, Jan. 2011.

Validating Automated Triggers and Notifications @ Scale in Blackboard Learn

John Whitmer, Ed.D.
Blackboard, Inc.
58 Maiden Lane, 5th floor
San Francisco, CA 94108
+1(530)554-1528
john.whitmer@blackboard.com

Sasha Dietrichson, Ph.D.
Blackboard, Inc.
1111 19th Street, NW
Washington, DC 20036
+1(800) 424-9299
sasha.dietrichson@blackboard.com

Bryan O'Haver
Blackboard, Inc.
190 W. Ostend Street, Suite 205
Baltimore, MD 21230
+1(800) 424-9299
bryan.ohaver@blackboard.com

ABSTRACT

Prior research on individual courses has demonstrated a significant relationship between use of the Learning Management System (LMS) and student course grade. Blackboard has created rule-based algorithms in a new LMS interface to notify students and faculty of students who may be at risk based on relative activity and grades received, and recognize positive behavior and grade achievement. This research project investigated the relationships underlying these algorithms against a large data set of LMS activity (1.2M student course weeks, 34,519 courses, 788 institutions). Findings included a small effect size in the relationship between time spent in the LMS and student grade; however, a small set of courses had a strong relationship that merits further research and consideration.

Keywords

Learning Analytics, Student Persistence, Student Retention, Higher Education, Learning Management Systems, LMS

1. INTRODUCTION

Multiple research studies on individual courses have found a significant relationship between use of the LMS and student grade [8, 7, 2, 3, 9, 10]. The value of LMS data in these courses has been larger than what is found in conventional demographic or academic experience variables in explaining variation in course grades. However, when analysis is expanded to all courses at an institution, several studies have found no relationship or an extremely small effect size in this relationship [1, 5, 4]. Does Learning Analytics only apply to only a small number of courses, or is it broadly applicable? What is the magnitude of this relationship, and is sufficiently large to include algorithms based on this relationship as a core functionality in academic technology platforms?

This question is of great practical significance for academic technology providers. Analytics functionality has typically been provided through custom data warehouses and analytics tools that include multiple data sources and systems, with custom integrations and algorithms. While useful and with accuracy that can be proven, these applications require significant resources to create and maintain, whether procured from a vendor or built in-house. They also require significant time to implement and deploy.

As part of Blackboard's new "Ultra" LMS course interface, rule-based triggers and notifications were created. For example, these rules would analyze course use and send the student and instructor a notification if a student's LMS activity dropped more than 10% from one week to the next. In addition to alerts of potentially at-risk students, positive encouragement alerts were also created to recognize outstanding achievement relative to self and others in the same course.

The rules were created based in prior research findings and an initial small data sample. However, additional validation with a larger data sample was required to ensure that the rules were meaningful predictors of student grade. This poster presents findings from this research on the question of accuracy and draws broader conclusions about the potential utility and generalizability of LMS activity data.

2. DATA SET AND ANALYSIS

The data analyzed for this project was sampled from log data recorded by Blackboard Learn. These logs were transformed into normalized data sets, and calculations made to estimate duration of time spent in the LMS by calculating the difference between start end end times for sessions. The data was aggregated at the institution-course-week-user level (e.g. one row per user per week per course per institution). The data sample included a complete set of students active for each course week, but did not include all weeks for each course. Each row also contained final course duration and final grade. A z-score of duration was calculated to provide a course-specific measure of student activity.

Given the importance of analyzing grade triggers and the relationship between activity and grade, only course-weeks with a graded entry for that week were included in the sample. Further, students with no activity have no logs and are therefore missing. This biases the sample toward courses making more intensive use of the LMS than a random sample.

Exploratory data analysis revealed a large number of rows with invalid grades and duration. The data was filtered to include courses with valid data and a potential for instructional use, namely: grade range between 0% and 120%, a minimum of 60 average minutes in the course, and a maximum of 5,040 minutes in the course per week, and enrollment more than 10 student and less than 500 students.

The final data set analyzed had the following profile:

Table 1. Data Set Characteristics

Records	Courses	Institutions
1.2M	34,591	788

Exploratory data analysis and distributions were conducted to ensure that the data was normally distributed and ensure other assumptions required for linear regression analysis were met. A

linear regression of final course grade on course duration and a logistic regression of course pass/fail on duration was run. Next, a separate linear regression was run for each course.

3. FINDINGS

As indicated in the scatterplot in Figure 1, there was a significant relationship between duration and grade. However, the effect size was extremely small (adjusted R^2=0.01537). Further, most of this effect was created by the intercept value; the coefficient for duration was 5.74e-04. Converted into practical effect, this coefficient indicates that for each additional hour spent in the LMS, students would gain 0.034% in their final course grade. Using course-relative measures of duration (e.g. z-scores by course) only increased the effect slightly (R^2=0.017). Logistic regression led to similar results.

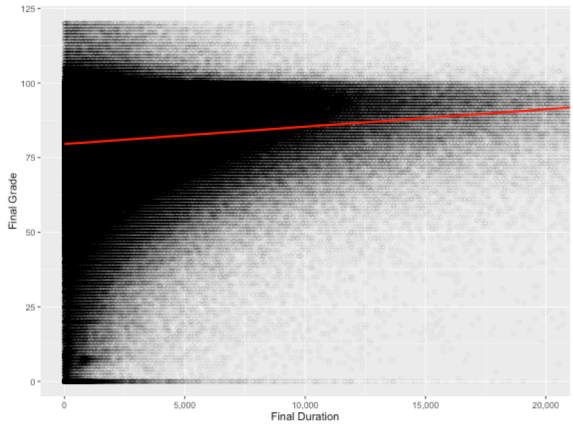


Figure 1. Duration vs. Grade across all Courses

When this regression was re-run at the course level, a high variation in this effect size was found. There were 7,648 (22%) courses with $p < 0.05$; the distribution in effect size is plotted below. Although skewed toward low values, there are a substantial number of courses with low to moderate effect sizes.

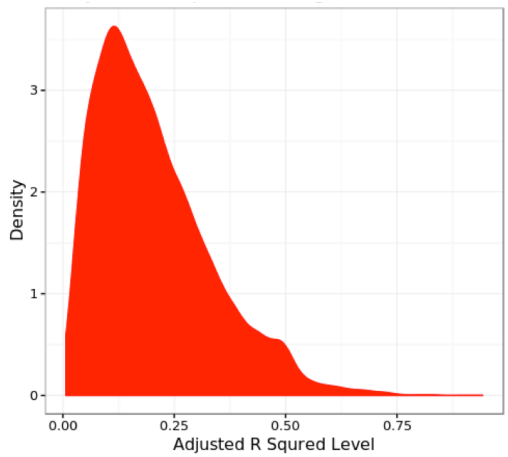


Figure 2. Adjusted R^2 Levels for Courses with Significant Duration vs. Grade Regressions

Initial data subsetting by available criteria (e.g. enrollment size, institution, average activity) did not identify a factor strongly related to this difference in effect.

4. IMPLICATIONS

These findings indicate that while rule-based triggers may not be predictive of student course achievement for all LMS courses, they are predictive for a substantial number of courses. Given known variability in how the LMS is used for instruction, these results provide an encouraging indication of potential value in this data. However, the reasons for this strong relationship among some courses and not among others is an important area for further research. We anticipate investigating issues in course design and early participation as identifiers of higher effect size.

As a result of this research, multiple modifications to the existing triggers in Blackboard Ultra have been made to refine and reduce the number of notifications sent. Further, a new configuration setting will be provided to disable these algorithms by course.

5. REFERENCES

[1] Campbell, J. P. (2007). Utilizing student data within the course management system to determine undergraduate student academic success: An exploratory study. (Educational Studies Ph.D.).

[2] Dawson, S., & McWilliam, E. (2008). Investigating the application of IT generated data as an indicator of learning and teaching performance (pp. 45). ASCILITE 2008, Melbourne.

[3] Fritz, J. (2011). Classroom walls that talk: Using online course activity data of successful students to raise self-awareness of underperforming peers. The Internet and Higher Education, 14(2), 89-97. doi: 10.1016/j.iheduc.2010.07.007

[4] Gašević, D., Dawson, S., Rogers, T., & Gasevic, D. (2016). Learning analytics should not promote one size fits all: The effects of instructional conditions in predicting academic success. The Internet and Higher Education, 28(January 2016), 68-84. doi: http://dx.doi.org/10.1016/j.iheduc.2015.10.002

[5] Lauria, E. J. M. B., Joshua. (2015, October 29-30, 2015). Mining Sakai to Measure Student Performance: Opportunities and Challenges in Academic Analytics. Paper presented at the European Conference on e-learning, Hatsfield, UK.

[6] Macfadyen, L. P., & Dawson, S. (2010). Mining LMS data to develop an "early warning system" for educators: A Proof of Concept. Computers & Education(54), 11.

[7] Morris, L. V., Finnegan, C., & Wu, S.-S. (2005). Tracking student behavior, persistence, and achievement in online courses. The Internet and Higher Education, 8(3), 221-231. doi: 10.1016/j.iheduc.2005.06.009

[8] Rafaeli, S., & Ravid, G. (1997). OnLine, Web Based Learning Environment for an Information Systems course: Access logs, Linearity and Performance. Paper presented at the ISECON 1997, Orlando, FL.

[9] Ryabov, I. (2012). The Effect of Time Online on Grades in Online Sociology Courses. MERLOT Journal of Online Learning and Teaching, 8(1).

[10] Whitmer, J., Fernandes, K., & Allen, B. (2012). Analytics in Progress: Technology Use, Student Characteristics, and Student Achievement. EDUCAUSE Review Online (July 2012).

Discovering 'Tough Love' Interventions Despite Dropout

Joseph Jay Williams
Harvard University
125 Mt Auburn St
Cambridge, MA 02138
joseph_jay_williams@harvard.edu

Anthony Botelho
Worcester Polytechnic Institute
100 Institute Road
Worcester, MA 01609 USA
abotelho@wpi.edu

Adam Sales
University of Texas
Statistics
Austin, TX 78712
asales@utexas.edu

Neil Heffernan
Worcester Polytechnic Institute
100 Institute Road
Worcester, MA 01609 USA
nth@wpi.edu

Charles Lang
New York University
239 Greene St
New York, NY, 10003
charles.lang@nyu.edu

ABSTRACT

This paper reports an application to educational intervention of Principal Stratification, a statistical method for estimating the effect of a treatment even when there are different rates of dropout in experimental and control conditions. We consider the potential value for using principal stratification to identify "Tough Love Interventions" – interventions that have a large effect but also increase the propensity of students to drop out. This method allowed us to generate an estimate of the treatment effect in an RCT without the selection bias induced by differential attrition by restricting analysis to just the inferred "stratum" of students who would not drop out in either condition. This paper provides a case study of how to appropriate the method of principal stratification from statistics and medical research fields to educational data mining, where it has been largely absent despite increasing relevance to online learning.

Keywords

principal stratification; selection bias; statistics; attrition; noncompliance; randomized controlled trial; experiment; online education

1. INTRODUCTION

A persistent problem in interpreting randomized experimental comparisons in learning environments is that the frequency of student dropout may vary between conditions . This is known as *differential attrition*, and causes problems with statistical inference [3] regarding the magnitude and direction of differences between treatment and control conditions. In cases where student completion is the metric of interest, such differences in condition are easily measured by the number of students to complete each; a problem arises, however, when performance is the metric of interest, as if less students drop out of one condition than the other, it is over-represented in the analysis causing unreliable results.

Differential attrition can mask the existence of what we label **"tough love"** interventions (TLIs). A TLI describes an intervention which introduces a treatment condition with features that cause some students to drop out, but has beneficial effects for students who persist. It is important to know *how much* such interventions impact a potential outcome in order to perform a cost-benefit comparison against the dropout rate. We believe that principal stratification is one tool that can be used to measure the effect of conditions in the presence of differential attrition and help identify TLIs.

2. ILLUSTRATIVE EXPERIMENT: IMPACT OF QUESTIONS ABOUT CONFIDENCE

In the preliminary data presented here, we consider a randomized controlled experiment (RCE) conducted within ASSISTments, a K-12 online and blended learning platform, reported in EDM 2015 [4]. Students were randomly assigned to either a condition of Treatment, where students were asked about their confidence in solving problems, or Control, where students were asked about technology usage. The data set used for analysis consists of 712 12-14 year olds in the eighth grade of a school district in the North East of the United States with 5,861 log records collected while students were solving math problems. The goal here is to estimate how the conditions differ in their impact on Mastery Speed, the number of problems needed to reach three consecutive correct responses indicating a sufficient level of understanding. It is important to note that a lower value in this metric indicates better performance.

3. ANALYTIC STRATEGY

Principal stratification [2, 5] is an approach to modeling causal effects for a subset of subjects defined subsequently to treatment assignment. For instance, it applies when issues of noncompliance, censoring-by-death, and surrogate outcomes within conditions have occurred. It uses two models, labeled here as the *Attrition* and *Outcome* models, to first stratify students and then estimate effects on a single stratum. Our

Attition model identifies four strata based on a student's likelihood to attrite: 1) **AA or Always Attriters:** Students who drop out regardless of condition. 2) **AC**: Students who complete if assigned to Treatment but drop out if assigned to control group. 3) **CA:** Students who only complete if assigned to Control. 4) **CC or "Never-Attriters":** Students who always complete regardless of condition; this is the stratum of interest for our work here, as it is the only group for which a treatment effect is well-defined.

True stratum membership is never observed, but must be inferred by the Attrition model using observed covariates, for which this work uses only the student's prior percent correctness labeled as acc_i. As attrition for one condition is known for each student, only the likelihood that the student would complete the opposing condition is inferred as seen in the following equations:

$$logit(Pr(completes_{i,ctrl} = 1)) = \alpha_{ctrl} + \beta_{ctrl} * acc_i$$

$$logit(Pr(completes_{i,treat} = 1)) = \alpha_{treat} + \beta_{treat} * acc_i$$

The Outcome model then observes only students placed in to the "Never-Attriter" stratum to estimate treatment effects. The equation used here utilizes the same covariates as the Attrition model with the addition of a dichotomized value of condition and a class-level variance term:

$$masteryspeed_i = \beta_{0s} + \beta_{1s} * acc_i + \beta_2 * cond_i + \sigma_i$$

The model parameters were estimated with Markov Chain Monte Carlo (MCMC) using four chains over 16000 iterations of which the first 8000 are omitted as a burn-in period allowing for convergence. The *Rhat* value shown in Table 1 reflects the degree of convergence of the Markov Chains, with the values near 1 indicating proper convergence. The results of the analysis are also seen in that table, and indicate that a TLI is not found as the effects of condition are not significant, falling within the confidence interval.

	mean	sd	0.95 CI	Rhat
Constant	1.78	0.13	(1.52,2.04)	1
Prior_Percent_Correct	-0.14	0.18	(-0.49,0.21)	1
Treatment	0.02	0.05	(-0.08,0.11)	1

	mean	sd	0.95 CI	Rhat
Constant	2.95	0.31	(2.34,3.55)	1
Prior_Percent_Correct	-1.33	0.39	(-2.09,-0.56)	1
Treatment	0.02	0.06	(-0.1,0.14)	1

Table 1: Typical Analysis: Coefficients for outcome model that predicts Mastery Speed based on Condition and Prior Accuracy, without using principal stratification (top) versus those coefficients using principal stratification (bottom).

4. SIMULATION STUDY

As no significance was found for coefficients in either case, a further comparison of principal stratification to traditional methods was conducted to verify principal stratification is beneficial in identifying such interventions when ground truth is known. The data generating model was designed to capture a tough-love intervention in which reliable difference could be found between conditions for students who would never drop out. For each simulated student, we assumed two latent/unobserved variables, intended to capture notions of *Grit* and *Ability*. There were two observed covariates, *prior percent complete*, which was a function of grit, and *prior percent correct*, which was a function of ability. The Outcome Variable (which might correspond to a post-homework quiz score) was a continuous variable that was a linear function of Ability.

A similar methodology to that described on the non-simulated dataset was then conducted. The coefficient for condition gave us a treatment effect for the never-attritor stratum. For comparison, we also conducted a Typical Analysis that estimated a treatment effect using ordinary least squares regression on all the data *without* using principal stratification and after 500 runs of the simulation, the 95% confidence interval from OLS included the average treatment effect for the never-attritors 62% of the time. In contrast, the principal stratification credible intervals were more efficient/reliable, including the true treatment effect 91% of the time.

5. CONCLUSION

This paper presented an explanation and case study application of principal stratification, to illustrate its potential as a method for analyzing randomized experiments and interventions in digital learning environments. One example from our analysis was identifying "Tough Love Interventions", but differential attrition pose a wide range of challenges to analyzing data from experiments, especially as learners gain flexibility in online environments such as Massive Open Online Courses (MOOCs). This makes the reliable analysis of experiments with variable dropout and attrition of increasing importance to the educational data mining community.

6. ACKNOWLEDGMENTS

This work is partially supported by the United States National Science Foundation Grant #DRL-1420374 to the RAND Corporation.

7. REFERENCES

[1] J. D. Angrist, G. W. Imbens, and D. B. Rubin. Identification of causal effects using instrumental variables. *Journal of the American statistical Association*, 91(434):444–455, 1996.

[2] C. E. Frangakis and D. B. Rubin. Principal stratification in causal inference. *Biometrics*, 58(1):21–29, 2002.

[3] J. Heckman, N. Hohmann, J. Smith, and M. Khoo. Substitution and dropout bias in social experiments: A study of an influential social experiment. *Quarterly Journal of Economics*, pages 651–694, 2000.

[4] C. Lang, N. Heffernan, K. Ostrow, and Y. Wang. The impact of incorporating student confidence items into an intelligent tutor: A randomized controlled trial. In *Proceedings of the 8th International Conference on Educational Data Mining*, 2015.

[5] A. C. Sales and J. F. Pane. Exploring causal mechanisms in a randomized effectiveness trial of the cognitive tutor. In *Proceedings of the 8th International Conference on Educational Data Mining*, 2015.

Stimulating collaborative activity in online social learning environments with Markov decision processes

Matthew Yee-King
Computing, Goldsmiths
University of London
m.yee-king@gold.ac.uk

Mark d'Inverno
Computing, Goldsmiths
University of London
dinverno@gold.ac.uk

ABSTRACT

Our work is motivated by a belief that social learning, where a community of students interact with eachother to co-create and share knowledge, is key to our students developing 21st century skills. However, convincing students to engage in and value this kind of activity is challenging. In this paper, we employ a technique from AI research called a Markov Decision Process (MDP) to model social learning activity then to suggest interventions that might increase the activity. We describe the model and its validation in simulation and draw conclusions about the effectiveness of this approach in general. The main contributions of the paper is to (i) show how it is possible to model education data as an MDP (ii) show that the resulting decision policy succeeds in guiding the system towards goal states in simulation.

Keywords

Social learning; Education system modelling, MDP, MOOC

Categories and Subject Descriptors

K.3.1 [**Collaborative learning**]: K.3.2 Computer science education G.3 Markov processes

1. INTRODUCTION

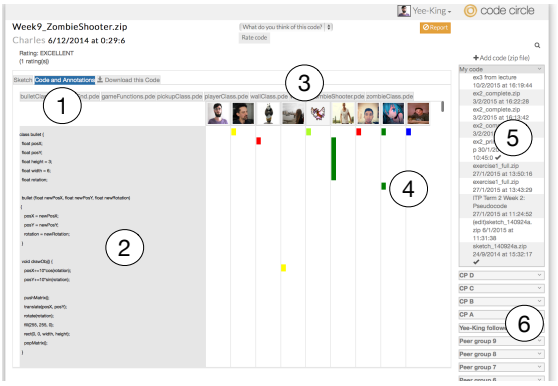
In this paper, we use a Markov Decision Process (MDP) to model social learning activity in terms of content consumption and content creation. This allows us to derive an 'action policy' which can potentially inform tutors and students what type of content to create and when to create it in order to maximise the levels of consumption of content in a social learning system. MDPs [2] are a commonly used method for sequential decision making under uncertainty, and they have been used in education technology e.g. [1]. The work presented here represents a novel application of MDP in a social learning context[1].

[1]A full version of the paper can be found at http://dx.doi.org/10.13140/RG.2.1.3592.0242

1.1 The case study and data set

The data used for the analysis presented here was collected during a 10 week case study involving 174 students on an introductory undergraduate programming course who were learning how to program using the Processing IDE. The students were using our social learning environment [3], as shown in Figure 1, which allow in-browser execution of programs as well as sharing, commenting and replying to comments on specific sections of code.

Figure 1: **The code discussion UI. 1) mode buttons: view running program, view code, download code, 2) the code viewer 3) the people who have commented on this code 4) a comment about a section of the code 5) my uploaded content 6) my communities.**



2. THE MODEL

MDP problems are formulated in terms of states, actions, state transitions, reward functions and action policies. The action policy dictates what is the best action to take in a given state in order to maximise future reward, where reward is defined in terms of the value of each state.

We begin by slicing the dataset into time windows and counting the number of activity types per window, split into content consumption and content creation activities. We define state as a 5 dimensional vector describing levels of 5 types of content consumption activity, namely read code, login, open thread, preview comment (pre-comm) and run code. The size of the state space is reduced by converting the raw

Table 1: **An excerpt from the action policy, showing its proposed content creation actions for the most commonly observed content consumption states**

read code	login	open thread	pre-comm	run code		comment	reply	share	grade-comm	grade code
0	0	0	0	0	→	0	0	0	0	1
1	0	0	0	0	→	0	0	0	0	1
2	2	2	2	2	→	2	0	0	0	0
1	0	0	1	1	→	2	0	0	0	0
2	1	2	2	2	→	0	0	2	0	0
State						Action				

log counts (e.g. number of times 'read code' happened) into 3 bands indicating low (0), medium (1) or high (2) activity relative to other time slices. For example, a state of 01012 would indicate low read code (0), medium login (1), low open thread (0), medium preview comment (1) and high run code (2). Some states are shown in Table 1. We define action as a 5 dimensional vector describing the levels of 5 content creation activities, namely comment, reply, share, grade comment (grade-comm) and grade code. As for state, they are reduced to low, medium or high relative to other time slices. For example, an action of 01012 would mean low comments, medium replies, low shares, medium comment grading, and high code grading (where we refer to the amount of grading, not the grades themselves). We can then gather observations of state-action-state tuples in the dataset, i.e. we observed this action being taken in this state and it was followed by this state. This is converted to a state transition matrix, an example entry in which is: "20020x00000" : "02002" : 0.5,"02222" : 0.5. In this example, state 20020 and action 00000 are observed to be followed by states 02002 and 02222 in an equal number of cases.

The next part of the MDP formulation is the reward function which involves assigning a value to every possible state plus a reward and cost for every possible state-action pair. State values are essentially sums of the elements of the state vector (state 00120 is worth 3). The values of state-action pairs are a sum of the values of all states observed to follow that state-action pair weighted by the number of observations of each follow on state. The cost for an action is calculated based on the frequency of that action in the observed set of actions, where we assume that infrequent actions are costly.

3. SIMULATION

Having derived an action policy, we will now evaluate it by runing it in simulation against a state transition matrix derived from a different period of the case study than the period the action policy was trained on. In this case, the training and test sets contained data derived from the same student cohort, just gathered during different time periods. The aim of this simulation is to examine the ability of the policy to generalise (to the same students in a different time window), and therefore to assess the potential usefulness of this system in a real world context.

Figure 2: **Test performance of the action policy in simulation vs. real world performance with varying time slice length. With 1 hour time slices, the MDP provides 1.25 times more value. Error bars are based on standard deviation over 100 simulation runs.**

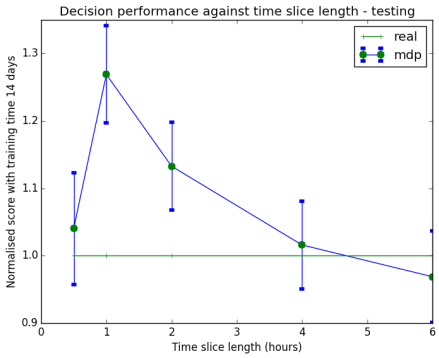


Figure 2 shows the results of running the action policy in simulation, where the training was carried out with varying time slice lengths. For each time slice length, the simulation was run 100 times to establish the typical range of performance. The one hour time slice provides the best performance, where the accumulated state values over the simulation were 1.25 times the value accumulated in the real case study data. It should be noted that the deteriorating performance as time slice length increases is likely to be caused by the smaller number of samples: there are less 6 hour slices than there are 1 hour slices. This means the transition matrix becomes very sparse, resulting in very limited simulation detail. This positive result demonstrates that the MDP approach could be a viable method to model and advise about online educational systems based around content consumption and creation.

4. CONCLUSION

We have described how social learning activity data can be formulated into an MDP and that this formulation allows the derivation of an action policy that can be used to decide what kind of content to create and when to create it, in order to maximise content consumption activity. We have also presented a preliminary validation of the action policy in a simulation based on real data, showing that the action policy selects actions that lead to higher levels of content consumption.

5. REFERENCES

[1] G. Durand, F. Laplante, and R. Kop. A Learning Design Recommendation System Based on Markov Decision Processes. In *17th ACM SIGKDD conference on knowledge discovery and data mining*, 2011.

[2] T. Snijders. Methods for longitudinal social network data: Review and Markov process models, 1995.

[3] M. Yee-King, M. Krivenski, H. Brenton, and M. D'Inverno. Designing educational social machines for effective feedback. In *8th International Conference on e-learning*, Lisbon, 2014. IADIS.

Predicting student grades from online, collaborative social learning metrics using K-NN

Matthew Yee-King
Computing, Goldsmiths
University of London
m.yee-king@gold.ac.uk

Andreu Grimalt-Reynes
Computing, Goldsmiths
University of London

Mark d'Inverno
Computing, Goldsmiths
University of London
dinverno@gold.ac.uk

ABSTRACT

We describe a collaborative video annotation system that aims to engage learners in a focused, collaborative process of content sharing and discussion, and explain how it was used in an online creative programming MOOC on Coursera. We explore the use of K-NN (K nearest neighbour) to predict which of a variable number of evenly spaced, final grade bands students will fall into based solely on a feature vector consisting of the total number of UI click and mouseover events they generated during the course. We were able to classify students into pass/fail bands with 88% precision; with 3 grade bands, precision was 77%, going down to 31% with 10 grade bands. Typically, a feature subset containing less than half of the available features provided the best performance.

Categories and Subject Descriptors

K.3.1 [**Collaborative learning**]: K.3.2 Computer science education

1. INTRODUCTION

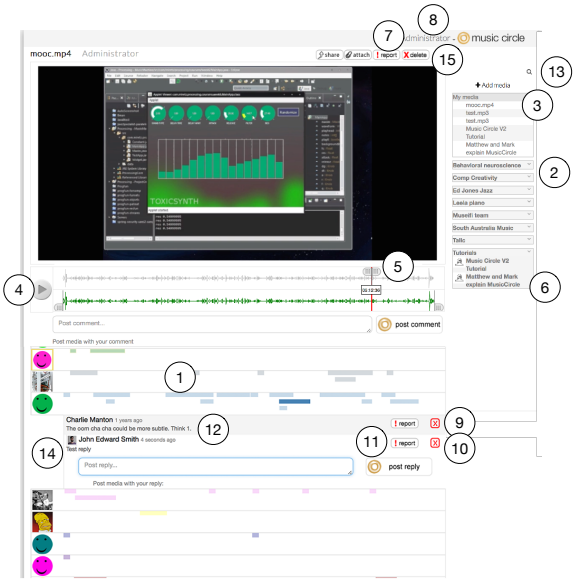
Our work is concerned with the development and analysis of systems that enable online, collaborative learning driven by different types of feedback. In this paper, we show how it is possible to predict student grades using user interface telemetry data gathered from a case study involving 993 students who completed all assessments for a creative programming course on MOOC platform Coursera. The students used a collaborative video annotation tool as part of their peer assessment, which we developed as part of an EU funded research project [5]. Previous work with collaborative media annotation systems and grade prediction includes [1, 3] and [2, 4] respectively[1].

2. THE CASE STUDY AND DATA SET

[1]A full version of the paper can be found at http://dx.doi.org/10.13140/RG.2.1.4525.9129

Three times during the course, the students were set a graded peer assessment wherein they had to extend our example programs and create a 5 minute video of themselves explaining their code and running their program. The videos were uploaded to our collaborative video annotation system wherein they could look at each others' videos and create annotations along a 'social timeline'. The system logged click and mouseover events on the UI elements shown in Figure 1, 3,716 unique users logged into our system. Of these, 3558 viewed one or more videos, 827 made one or more comments, and 258 made one or more replies to comments. 2,898 videos were submitted for three separate assessments, and were viewed a total of 112,189 times. 7,370 comments were made, and 978 replies. For this paper, we filtered the data down to all logged click and mouseover events for students who gained a final grade on the course, a total of 993 students.

Figure 1: **A screen shot of the video annotation and discussion system. The numbered labels show all of the elements of the UI for which events are logged automatically.**

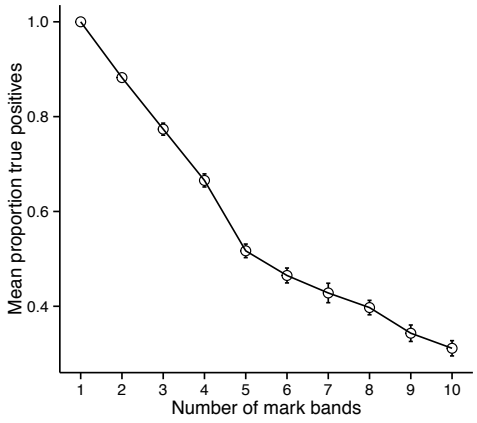


3. ANALYSIS

To predict student grades, we created a 16 dimension feature vector consisting of total numbers of clicks and mouseover events on each of the GUI elements shown in Figure 1 plus the final grade achieved by the student. We began by attempting to correlate individual elements of the feature vector to the final grade but individual correlations were too weak to predict grades, ranging between 0.53 and 0.18. This motivated us to try a multivariate classification approach. For our first analysis, we assigned labels to the students based on which of N evenly spaced grade boundaries they fell into. For example, if $N = 2$, then students were labelled **1** if $final\_grade < 50$ and **2** if $final\_grade \geq 50$. We split the dataset into equally sized training and test sets and attempted to train a K-NN classifier to assign labels to the test set, with varying numbers of mark bands and multiple run cross validation.

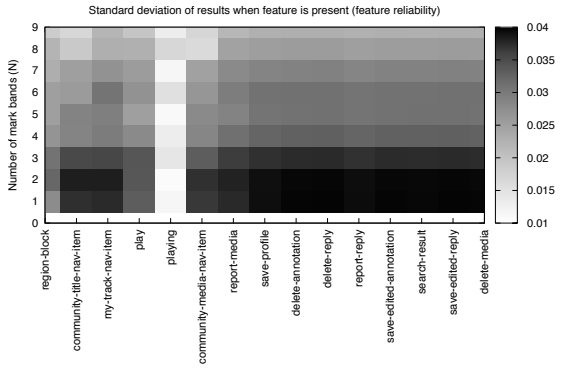
Figure 2 shows the proportion of correctly assigned labels in the test set as number of mark bands N varies from 1 to 10. For example, the pass/fail classification where $N = 2$ achieved 88% true positives. We note that the distribution of marks across the bands has a significant impact on the meaning of accuracy, and that for $N = 2$, for example, there are a large number of examples in each class which are being correctly classified.

Figure 2: **Performance of the classifier with $k = 6$ and number of mark bands $N = 1...10$.**



For our second analysis, we tried out all possible combinations of feature elements to see which combination achieved the highest classification accuracy. Since the number of features was 15, the number of permutations was 32768 (2^15). K was set to 6 and number of mark bands N varied from 1-10. Figure 3 highlights the most reliable features in the feature set by showing how much the prediction score varied (the standard deviation) across the set of all permutations per N which involved that specific feature. To be clear, it does not differentiate between features that reliably provide good or bad results. The most reliable feature was 'playing', which is triggered automatically while a video is playing. The second most reliable feature was 'region block', which is logged when a user clicks on a comment on the timeline to open the discussion thread. More work is needed to understand this result more deeply.

Figure 3: **Heat plot showing the standard deviation in the prediction results when different features are present. Low variation (lighter) is desirable, meaning a feature makes a reliable contribution to the results.**



4. CONCLUSION

We have briefly described a collaborative video annotation tool we have developed. Using interface telemetry data gathered describing click and mouseover events generated by the user interface of the system, we were able use a K-NN classifier to classify students into pass/fail bands with 88% precision; with 3 grade bands, precision was 77%, and with 10 bands it was 31%. We measured the prediction power of different combinations of the features and were able to identify the most reliable features, which relate to playing back videos, exploring content menus and reading comments.

5. REFERENCES

[1] D. Bargeron and J. Grudin. Asynchronous Collaboration Around Multimedia Applied to On-Demand Education. *Journal of Management Information Systems*, 18(4):117—-145, 2002.

[2] C. a. Coleman, D. T. Seaton, and I. Chuang. Probabilistic Use Cases: Discovering Behavioral Patterns for Predicting Certification. *Proceedings of the Second (2015) ACM Conference on Learning @ Scale - L@S '15*, pages 141–148, 2015.

[3] E. F. Risko, T. Foulsham, S. Dawson, and A. Kingstone. (CLAS): A New TOOL for Distributed Learning. 6(1):4–13, 2013.

[4] T. Sinha, P. Jermann, N. Li, and P. Dillenbourg. Your click decides your fate: Leveraging clickstream patterns in MOOC videos to infer students' information processing and attrition behavior. 2014.

[5] M. Yee-King, M. Krivenski, H. Brenton, and M. D'Inverno. Designing educational social machines for effective feedback. In *8th International Conference on e-learning*, Lisbon, 2014. IADIS.

Meta-learning for predicting the best vote aggregation method: Case study in collaborative searching of LOs

Alfredo Zapata[1], Victor H. Menéndez[1], Cristóbal Romero[2], Manuel. E. Prieto[3]
[1]Autonomous University of Yucatan, Faculty of Education, 97305, Mérida, Mexico
[2]University of Cordoba, Dept. of Computer Science, 14071, Córdoba, Spain
[3]University of Castilla-La Mancha, Computer Science Faculty, 13071, Ciudad Real, Spain
{zgonza, mdoming}@correo.uady.mx, cromero@uco.es, manuel.prieto@uclm.es

ABSTRACT
The problem of recommending learning objects to a group of users or instructors is much more difficult than the traditional problem of recommending to only one individual. To resolve this problem, this paper proposes to use meta-learning for predicting the best voting aggregation strategy in order to automatically obtain the final ratings without having to reach a consensus between all the instructors. We have carried out an experiment using data from 50 groups of instructors doing a collaborative search of LOs in AGORA repository.

Keywords
Meta-learning, Classification, LOs Collaborative Search

1. INTRODUCTION
Nowadays, there is a wide variety of e-learning repositories that provide digital resources for education in the form of Learning Objetcs (LOs). The search for and recommendation of LOs are traditionally viewed as a solitary and individual task but this is changing. On the one hand, collaborative search can be more effective than an individual search, for example in our case, a group of instructors may be interested in searching and selecting together the educational resources most appropriate to develop a new digital course. On the other hand, the goal of group recommendation is to compute a recommendation score for each item (in our case, each LO) that reflects the interests and preferences of all group members. The problem is that all group members may not always have the same tastes, and a consensus score for each item needs to be carefully designed. So, to recommend to user groups is more complicated than recommending to individuals [2]. The main problem that group recommendation needs to solve is how to adapt to the group as a whole, based on information about individual users' likes and dislikes. A solution is to use group decision strategies or aggregation methods that are inspired by social choice theory, and establish different automatic ways of how a group of people can reach a consensus. However, groups are very diverse, and no single group decision strategy works best for all groups. A way to address this issue is to identify the inherent characteristics of different groups and to determine their impacts on the group decision process [1]. Following this idea, in this paper we propose to use meta-learning for predicting the best aggregation method recommended for a group based on its characteristics. In this way, the traditional time-consuming consensus-taking among users can be avoided by using an automatic method based on meta-learning.

2. PROPOSED METHODOLOGY
In order to resolve the problem of determining which aggregation method is the most appropriate for each type of collaborative search group, we propose to use a meta-learning process (Fig. 1). The idea is to obtain automatically the aggregation method which provided/gave the best performance for a group of instructors based on its characteristics and previous rating of other similar groups. As seen in Fig. 1, the meta-learning process starts from a dataset which contains descriptive information about groups, the individual ratings of each member to all the LO's selected by the group during the collaborative search, and the consensus about the final rating assigned to all selected LO's. Next, the groups' characteristics are defined and the performance of the rating aggregation methods is evaluated in order to form a new metadata set. Then we select a classification algorithm that it used each time we have a new group of users/instructors in order to can recommend an aggregation method of their LO's rating.

Firstly, in order to create metadata, we use the following previously proposed descriptors or characteristics [1]: group size, social contact level, experience level and dissimilarity level. Additionally, we also propose a new descriptor based on the activity level of the group members in using LO repositories. Then, an evaluation phase is necessary in order to determine which aggregation method obtains the lowest error with respect to the actual consensual final rating of group members for all LOs. This actual or real rating is the final score of the group, obtained after consensus between all the members. So, it is necessary that the group have an in-person reunion or online communication in order to achieve the final score, starting with each individual rating/score and opinion. Various aggregation methods can be used to automatically obtain the final group rating for each LO [2]. We propose to use eight traditional aggregation methods

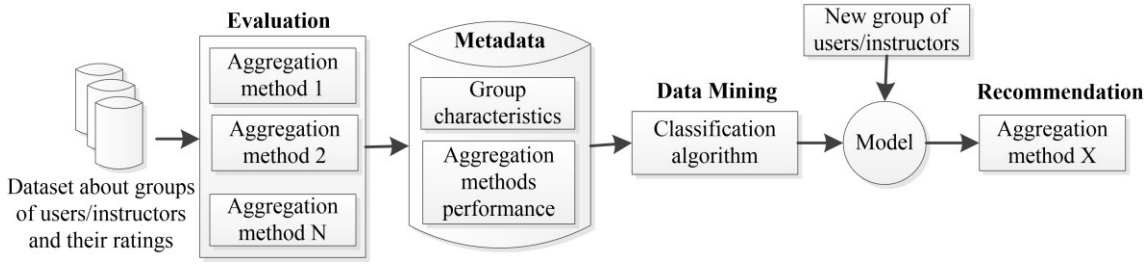


Figure 1. Meta-learning process for recommending a voting aggregation method.

(plurality voting, average, median, approval voting, least misery, most pleasure, average without misery, and fairness) plus three new weighted versions (active, social and experience user) of the average method based on [3]. In our case, instead of assuming equal weights for all the members, we give more weight to some users based on their characteristics, assuming that some members are more influential and can persuade others to agree with them. Next, a new metadata set is created by using both the characteristics of each group and the obtained aggregation method that provided the best group performance. After that, a classification algorithm is used to predict which aggregation method is most appropriate for a new group, given its characteristics. However, because there are a lot of classification techniques, we must therefore select a representative number of classification algorithms in order to compare their performance when using our metadata set. Finally, the classification algorithm that provides a better general performance will be the one selected for predicting the aggregation method most appropriate for each new group. In this way, the classification model obtained by the selected algorithm will be used for selecting, in real time, the best aggregation method for a new group according to the characteristics of the group and their individual ratings.

3. EXPERIMENTAL WORK

We have carried out an experiment in order to test our proposal of predicting the most appropriate aggregation method to use with a new group, based on the characteristics of the group members and the previous rating of similar groups. We have used data from a collaborative search of LOs in DELPHOS system [5]. We sent invitations, without using any incentive, to all instructors and final-year students of the Faculty of Education of the Autonomous University of Yucatan in Mexico to participate in the experiment. Only 75 users accepted our invitation: 27 professors or university teachers at different levels (assistant, associate and full) and 48 final-year students. We defined a total of 50 different groups of instructors and students with different typologies on their characteristics. We created a metadata set that contains both the previous characteristics/descriptors of the 50 groups as well as the best aggregation methods for each group by evaluating the performance of the 11 used rating aggregation strategies (see Table 1). In order to do this, we have used RMSE (Root-Mean-Square Error) of each aggregation method in each group. Starting from this metadata set, it is possible to predict the best aggregation method to a new group by using a classification algorithm. This is a classification in which the class or attribute to predict is precisely the aggregation method that obtains the best ranking. To this end, we have used different classification algorithms provided by the WEKA software, which is one of the most popular and most used tools for data mining. We have selected a representative number of the best known classification algorithms available in WEKA: JRip (implementation of RIPPER algorithm), J48 (implementation of C4.5 algorithm), NaiveBayesSimple (implementation of Bayes classifier), SMO (implementation of support vector classifier) and IBk (implementation of KNN or Nearest Neighbours algorithm). We have executed the previous five classification algorithms using their default parameter values and 10-fold cross-validation. In order to evaluate the classification performance and to determine the best algorithm for each group, we have used two measures that have previously been used to evaluate classification algorithm recommendation methods [4]. The first is called ARE (Average Recommendation Error) and it measures the average error of the current recommendation (predicted aggregation method) regarding the best and the worst recommendation (best and worst aggregation methods from the list of methods ordered from the lowest to the highest RMSE). The second measure is the Reciprocal Average Hit Rate, also known as Mean Reciprocal Rank (MRR), which measures the median position occupied by the method currently predicted for each of the groups in the complete list of methods ordered by RMSE.

Table 1. Average Recommendation Error and Mean Reciprocal Rank obtained by the 5 classification algorithms.

Algorithm	ARE	MRR
IBk	0,9418	0,3506
J48	0,9492	0,4239
JRIP	0,9594	0,5453
NaiveBayes	0,9458	0,4113
SMO	0,9583	0,4689

As we can see in Table 1, IBk was the best classification/prediction algorithm (followed by NaiveBayes and J48) because it obtained the lowest value of Average Recommendation Error and the lowest value of Mean Reciprocal Rank. So, since the algorithm IBk achieved the best results, it is our selected classification algorithm to automatically recommend the best aggregation method of the most similar group or nearest neighbours to every new group as the best method for rating all the LOs added to the group. In this way, the moderator of the group would use the recommended aggregation method obtained by the IBk algorithm instead of having to conduct the traditional consensual decision process.

4. ACKNOWLEDGMENTS

The authors gratefully acknowledge the financial subsidy provided by Spanish Ministry of Science and Technology TIN2014-55252-P.

5. REFERENCES

[1] Gartrell, M., Xing, X., Lv, Q., Beach, A., Han, R., Mishra, S., Seada, K., 2010. Enhancing group recommendation by incorporating social relationship interactions. In: *Proceedings of the 16th ACM GROUP '10*, ACM Press, New York, NY, USA, pp. 97–106.

[2] Masthoff, J., 2011. Group recommender systems: combining individual models, in: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (Eds.), *Recommender Systems Handbook*. Springer Press, New York, pp. 677–702.

[3] Popescu, G., 2013. Group recommender systems as a voting problem. In: *Proceedings of the 5th International Conference on Online Communities and Social Computing*, OCSC 2013, Springer, Berlin, pp. 412–421.

[4] Song, Q., Wang, G., Wang, C., 2012. Automatic recommendation of classification algorithms based on data set characteristics. *Pattern Recogn*. 45(7), 2672–2689.

[5] Zapata, A., Menéndez, V.H., Prieto, M.E., Romero, C., 2013. A framework for recommendation in learning object repositories: an example of application in civil engineering. Adv. Eng. Softw. 56, 1–14.

Soft Clustering of Physics Misconceptions Using a Mixed Membership Model

Guoguo Zheng
University of Georgia
Athens, GA
ggzheng@uga.edu

Seohyun Kim
University of Georgia
Athens, GA
seohyun@uga.edu

Yanyan Tan
University of Georgia
Athens, GA
yanyan.tan25@uga.edu

April Galyardt
University of Georgia
Athens, GA
galyardt@uga.edu

ABSTRACT

Students often possess multiple, conflicting misconceptions which may be activated and expressed in different contexts. In this paper, we use a mixed membership model to explore the patterns of misconceptions in introductory physics. Mixed membership models have been widely used for modeling observations that have partial membership in several latent groups. The latent groups in the current study are misconception patterns. This model allows us to examine whether students are likely to hold a few or many misconceptions, as well as which misconceptions are likely to co-exist. Physics knowledge was measured with the Force concepts inventory (FCI). We found three dominant response patterns, with different misconceptions prominent within each pattern.

1. INTRODUCTION

Student misconceptions can be persistent, and interfere with learning unless they are addressed directly. One important characteristic of misconceptions is that students possess many different knowledge components simultaneously, so that the particular schema or rule a student uses to solve a question depends on many different factors, including the context of the question [4]. This paper presents a case-study for using a mixed-membership model [1] to capture the characteristics and coherent patterns among students' misconceptions in introductory physics. Mixed membership model allows students to possess different misconception patterns (profile) across test questions. In this study, we focus on two questions: (1) What are the common misconception pattern students possess across the test, and which misconceptions tend to co-occur. (2) How much does each student exhibit each pattern?

2. METHODS

2.1 Mixed membership model

Mixed membership models allow an individual to switch profiles across contexts, test items. How much each individual uses each profile is parametrized by $\theta_i = (\theta_{i1}, \ldots, \theta_{iK})$. The components of θ_i are nonnegative and sum up to 1. Z_{ij} indicates the profile that student i uses for item j, so that

$$Z_{ij}|\theta_i \sim Multinomial(\theta_i).$$

Each latent profile has its own probability distribution for observed variables. Since the items from the case study are multiple choice, if X_{ij} denotes the observed response for student i on item j, then $X_{ij}|Z_{ij} = k \sim Multinomial(\beta_{(j|Z_{ij}=k)})$, where $\beta_{(j|Z_{ij}=k)} = (\beta_{kj1}, \ldots, \beta_{kjm}, \ldots, \beta_{kjM})$, β_{kjm} denotes the probability that a student using profile k on item j will select option m, and M is the number of options.

In the mixed membership model, the generative process is given by [5,6]:

1. For each item $j = 1, \ldots, J$, draw $\beta_{(j|Z=k)} \sim Dirichlet(\eta)$, for $k = 1, \ldots, K$.
2. For each individual $i = 1, \ldots, N$
   (a) Draw $\theta_i \sim Dirichlet(\alpha)$
   (b) For each item $j = 1, \ldots, J$,
      i. Draw $Z_{ij}|\theta_i \sim Multinomial(\theta_i)$.
      ii. Draw $X_{ij}|Z_{ij} \sim Multinomial(\beta_{(j|Z_{ij}=k)})$,

Here η and α are prior parameters. These could be estimated in an empirical-Bayes fashion. We choose to set these parameters to incorporate prior information, and stabilize the model.

2.2 FCI Data

From 1995-1999, 4450 high school students responded to The Force Concept Inventory (FCI), one of the most commonly used assessments in physics to measure students' understanding of concepts on Newtonian mechanics. We foucsed on the pre-test scores from a larger study [3]. The FCI consists of 30 multiple-choice items, with 18 items measuring *Newton's Second Law*. Most of the distractor options on this test were designed to map to a common physics misconception, though some distractors are statements that cannot be

explained by physics theories. More detailed explanation of these misconceptions can be found in [2].

3. RESULTS

We estimated the mixed membership model using MCMC with 5,000 iterations (1,000 burn-in). We placed a weakly informative prior on $\beta_{(j|Z=1)}$, of $\eta_{j1} = (50, 1, 1, 1, 1)$, and a flat prior to all the other parameters.

3.1 Number of Profiles

We fit mixed membership model with three to seven profiles. The same misconceptions were found to co-exist regardless of the number of profiles. In the 3-profile model, students have the most distinct probabilities of selecting a particular response across profiles, and were more likely to exclusively belong to one of the profiles ($\theta_{ik} > 0.8$). Thus, we can say that three profiles is representative of students' misconception patterns and in this paper, we focus on the 3-profile model.

3.2 Students' Membership in the Profiles

Profile memberhsip of each student is captured by the parameter $\theta_i = (\theta_{i1}, \theta_{i2}, \theta_{i3})$ shown in Figure 1. The proportion of students who exclusively belong to profile 3 is the highest, followed by profile 2 and profile 1. There are many students who are between profile 2 and profile 3 as well as between profile 3 and profile 1. Far fewer students fall between profile 2 and profile 1.

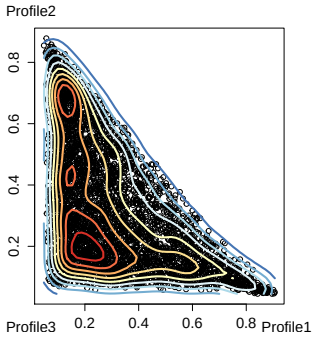


Figure 1: Contour map of posterior distribution for students' membership in the three profiles. X nad Y axes represent θ_{i1} for Profile 1 and Profile 2 (θ_{i1}) respectively. Profile 3 can be obtained by $\theta_{i3} = 1 = \theta_{1i} - \theta_{i2}$

3.3 Characteristics of Profiles

Each profile is parameterized by a probability distribution over the responses to each item, $\beta_{(j|Z=k)} = (\beta_{kj1}, \ldots, \beta_{kj5})$. We illustrate the characteristics of each profile using items that measure Newton's Second Law of Motion, and these characteristics hold up for all the items in the FCI instrument.

Misconception Profile (profile 3) This profile is characterized by high probability on responses containing misconceptions. Recall also, that this profile had the most students that belonged to it exclusively, as well as large numbers of students who were between it and the other profiles (Figure 1). In this profile, some misconceptions, such as *impetus dissipation* are observed repeatedly across items. However, we also observe that the activation of a misconception depends on items. For example, the misconception *impetus supplied by "hit"* is likely to be observed in item 30 even though it is also associated with item 11. This profile has the most profound implications for instruction since it is the largest, and demonstrates that students tend to not hold a single misconception, but rather many misconceptions that co-exist and may be expressed in different contexts.

Mostly Correct Profile (profile 1). This profile places a high probability on the correct response for most items, and has the smallest number of students that have high membership in the profile. However, on a few items, this profile is also associated with misconceptions. Some of these misconceptions, such as *largest force determines motion* were shared by the other profiles which instructors will want to address, and some of them tend to be of a higher-level.

Uniform Profile (profile 2). In general, the probability of choosing an option was similar across at least three options for most of the items. This profile has a large number of students who belong almost exclusively to it. Even when we increased the number of profiles, it did not disappear, nor decompose into separate profiles. These observations indicate that students in this profile do not have any coherent pattern in their responses.

4. CONCLUSION AND DISCUSSION

This study illustrates how mixed membership models can be a good tool to summarize a number of misconceptions into fewer numbers of profiles by identifying misconceptions that are likely to co-exist. Among the three profiles we found with FCI data, the majority of students had partial or complete membership in the *misconception profile*. The high coherence of co-existing misconceptions across a large number of students in this profile demonstrates the real power of this mixed membership analysis. By finding coherent patterns exhibited by many students at least some of the time, we find evidence that may suggest new theory. Future work can focus on the challenge of deciding an optimal number of profiles when conducting mixed membership models and the assumption that Z_{ij} depends on both i and j. Profile transitions between pre- and post-test should also be examined.

5. REFERENCES

[1] E. Erosheva, S. Fienberg, and J. Lafferty. Mixed-membership models of scientific publications. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5220–5227, 2004.

[2] D. Hestenes and J. Jackson.

[3] D. Hestenes, M. Wells, G. Swackhamer, et al. Force concept inventory. *The physics teacher*, 30(3):141–158, 1992.

[4] K. R. Koedinger, A. T. Corbett, and C. Perfetti. The knowledge-learning-instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive science*, 36(5):757–798, 2012.

Perfect Scores Indicate Good Students !? The Case of One Hundred Percenters in a Math Learning System

Zhilin Zheng
Department of Computer Science
Humboldt-Universität zu Berlin
Berlin, Germany
zhilin.zheng@hu-berlin.de

Martin Stapel
Department of Computer Science
Humboldt-Universität zu Berlin
Berlin, Germany
martin.stapel@hu-berlin.de

Niels Pinkwart
Department of Computer Science
Humboldt-Universität zu Berlin
Berlin, Germany
niels.pinkwart@hu-berlin.de

ABSTRACT
As a teacher or administrator, seeing a student scoring 100% in an exercise series within an online learning system would typically raise no immediate worries. This paper analyzes the "one hundred percenter" sessions in a math learning system. We argue that some student sessions with 100% score may actually not be predictive of student's learning success, and that a frequently exhibited student strategy of getting a perfect score by skipping exercises and repeating series is not ideal.

Keywords
Learning Analytics; Educational Data Mining; User Modelling; Student Behavior; Gamification

1. INTRODUCTION
Many educational technology systems allow students to take exercises multiple times and thus follow a resubmission policy [4; 6]. In this model, students have a chance to revise their answers by looking closely at their errors and the system gives feedback accordingly (which may vary in form and degree of detail). This resubmission policy certainly benefits self-regulated learning. Some of these learning systems limit the number of resubmissions, whereas others leave it unlimited [6]. Nevertheless, a possible negative side effect of this policy is evident as well. Under a resubmission policy, students can potentially take a trail-and-error strategy with little or even no thinking about the exercises and still try to get a high score [1; 4]. To address this issue, randomized initial data can be used to generate new (but structurally similar) exercises and thus avoiding repetitive occurrences of same exercises [5]. This strategy has shown to have a positive impact on students' learning results [6].

In this paper, we conduct an investigation in the context of a math learning system with a feature of resubmission. Log files indicate that a portion of students were eager to achieve a 100% success rate by taking a strategy of skipping exercises with a 'help' of resubmission. As far as we know, this phenomenon has not been studied extensively up to now. Nevertheless, skipping behavior itself is quite common in computer-supported learning systems. If a resubmission policy is allowed, restarting an exercise series or a quiz is technically possible and not as expensive as in paper-and-pencil tests in physical classroom settings. One may argue that students' motivation of achieving a 100% success is not surprising too. In a traditional classroom this happens quite often because students desire their teacher's praise or want to show off their talent with such a high learning performance. In this paper we thus do not primarily intend to discuss the phenomenon as such, but want to investigate two related questions. First, is this skipping strategy (aborting and restarting an exercise series after a mistake) actually a fast way to achieve a 100% success score, or are there more efficient strategies to reach this goal? Second, from a pedagogical viewpoint, do students who take this strategy perform as good as their learning outcomes seem to indicate – i.e., perfectly?

2. DATA
Bettermarks[1] is an online math learning system. It delivers math learning content in cooperation with K-12 schools (grades 4-10). Since the system provides flexibility to choose math topics and exercise series according to needs of different curriculums, it is frequently blended into classroom teaching by school teachers. Typically, teachers assign exercises (organized in exercise series) to their students and their achievement is in turn reported back to the teachers via the system. Bettermarks employs an unlimited resubmission strategy, which means that students can make as many attempts as they want. With such a feature, students are expected to iteratively make use of more attempts to correct their errors with helps of the system's feedback and/or hints.

After a close look at the sever log file, we found that plenty of the students made many skipping attempts before a 100% success. We termed such an interesting phenomenon as a "one hundred percenter with skipping". They did not take the exercises one after another as some of their peers did. Instead they skipped all the remaining exercises and made a new attempt once an error occurred. From January 2014 till November 2014 we found 8,640 (6.4%) sessions involved in such a phenomenon out of totally 687,688 sessions.

3. ANALYSES AND RESULTS
We identified another two different groups of student sessions with least one 100% success in one attempt of the exercise series. One group is the sessions without any skipping behavior but at least a 100% success once (59,941 in total). The other group contains sessions with a 100% success at the first attempt, but still with next attempts in the same exercise series. We termed this group "strong one hundred percenters" (3,854). The one hundred percenters with skipping showed a totally different learning style than their counterparts without skipping. Upon realizing a problem (e.g., a mistake made or an apparent difficult

[1] http://bettermarks.com/

exercise), the former group decided to skip over this exercise and the remaining ones in the series, and restarted the series. To the contrary, the ones without skipping chose to continue with the current work. They took every learning chance (as the system designer or the teacher would probably have hoped). Through this behavior, they could still probably learn something from the feedback or the next exercises in the series even though they had made an error. However, their desire to achieve a 100% success was evident through their behavior. The question which style (with or without skips) leads to the shared goal (100% success) quicker is interesting. To answer this question, we counted the students' attempts to a 100% success respectively. Students with the skipping strategy in fact needed more attempts to achieve their desired perfect score (3.6 attempts vs 2.4 attempts). This difference is statistically significant (Welch's t-test with different variance, $p<0.001$). In other words, students that chose to do all the exercises instead of skipping achieved a 100% success faster. Note that we took the number of attempts as a measure instead of time spent because that would bring individual's faster or slower learning pace as a noise into our analysis.

Interestingly, some of the one hundred percenters continued with their learning activities even after having obtained a perfect score. They even made more attempts right after their achievement of 100% success. In this case, we can hypothesize that the reward-oriented motivation was lower than the intrinsic, learning-oriented motivation: the system would reward students achievement badges once they achieved a 100% success but no more afterwards. We got 129 (1.4%) of such sessions out of the one hundred percenters with skipping, 1,414 (2.3%) sessions out of the one hundred percenters without skipping, and 3,854 (by definition, 100%) sessions out of the strong one hundred percenters. Solely from the participation we can intuitively see that very few one hundred percenters with skipping engaged in their learning activities once they had got the achievement badges in comparison of another two groups. We sought to investigate their learning performance under this situation (only with intrinsic motivation). We calculated their average success rate over attempts after that 100% success attempt. The average learning performance of one hundred percenters with skipping (0.78) is much lower than without skipping (0.91). Unsurprisingly, the strong one hundred percenters take the leading position (0.94). A Kruskal-Wallis H-test confirmed significant difference ($p<0.001$).

We can now give some answers to our questions stated in Section 1. First, the skipping strategy does not show any advantage when compared to the non-skipping strategy. To the contrary, students who take this strategy needed more attempts to achieve a 100% success at the end. More importantly, one hundred percenters with skipping reveal significantly weaker capabilities than their peers during the attempts after a 100% success. This would put this portion of students at risk especially when teachers only take their best outcome as a rating criterion. Since they do not show any weakness solely on that indicator, their teachers would overlook them (assuming they do fine) and move their attention to the weak students. As such, one hundred percenter behavior with skipping is not a fruitful strategy – it does not make the process of getting the 100% badge more efficient, and in fact students that pursue this strategy did not learn as much as their scores indicate, and less than their peers.

4. CONCLUSION

This work analyzes a portion of students in a math learning environment who achieve a 100% success in an exercise series through skipping exercises and then repeating the series. A closer look at the data in the learning system yielded several insights. The first one is that the adoption of the skipping strategy does not help to speed up to a 100% success. Instead, a non-skipping strategy leads students to achieve a perfect score faster. Another yet more important finding is that one hundred percenter behavior could put students at risk of being overlooked by teachers. They actually do not perform as excellent as their learning performance indicates.

With regard to the motivation of one hundred percenters, achievement badges available in the system, a gamification strategy often used in educational systems, could explain their motivation. Still there could be some other incentives, for example, encouragement or rewards coming from somewhere outside of the learning system. The learning system we studied is integrated into blended teaching settings in most cases. Thus teachers should have much space to motivate their students without a need to solely rely on the learning system's rewarding strategy. Apart from motivation factors, carelessness or a slip [2; 3] could explain one hundred percenters' skipping behavior as well.

5. ACKNOWLEDGMENTS

Our thanks to the Chinese Scholarship Council (CSC) for funding the first author's research.

6. REFERENCES

[1] AUVINEN, T., 2015. Harmful Study Habits in Online Learning Environments with Automatic Assessment. In *Learning and Teaching in Computing and Engineering (LaTiCE), 2015 International Conference on*, 50-57.

[2] BAKER, R.S., CORBETT, A.T., and ALEVEN, V., 2008. More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. In *Intelligent Tutoring Systems* Springer, 406-415.

[3] HERSHKOVITZ, A., HERSHKOVITZ, R.S.J., DE BAKER, J., GOBERT, M., WIXON, M.S., and PEDRO, 2013. Discovery With Models: A Case Study on Carelessness in Computer-Based Science Inquiry. *American Behavioral Scientist 57*, 10, 1480-1499.

[4] KARAVIRTA, V., KORHONEN, A., and MALMI, L., 2006. On the use of resubmissions in automatic assessment systems. *Computer Science Education 16*, 3 (2006/09/01), 229-240.

[5] KORHONEN, A. and MALMI, L., 2000. Algorithm simulation with automatic assessment. In *Proceedings of the Proceedings of the 5th annual SIGCSE/SIGCUE ITiCSEconference on Innovation and technology in computer science education* (Helsinki, Finland2000), ACM, 343157, 160-163.

[6] MALMI, L. and KORHONEN, A., 2004. Automatic feedback and resubmissions as learning aid. In *Advanced Learning Technologies, 2004. Proceedings. IEEE International Conference on*, 186-190.