

Mining Knowledge Components From Many Untagged Questions

Neil L Zimmerman
McGraw-Hill Education
281 Summer St
Boston, MA 02210
neil.zimmerman@mheducation.com

Ryan S Baker
Graduate School of Education
University of Pennsylvania
3700 Walnut Street
Philadelphia, PA 19104
rybaker@upenn.edu

ABSTRACT

An ongoing study is being run to ensure that the McGraw-Hill Education LearnSmart platform teaches students as efficiently as possible. The first step in doing so is to identify what Knowledge Components (KCs) exist in the content; while the content is tagged by experts, these tags need to be re-calibrated periodically.

LearnSmart courses are organized into chapters corresponding to those found in a textbook; each chapter can have anywhere from about a hundred to a few thousand questions. The KC extraction algorithms proposed by Barnes [1] and Desmarais et al [3] are applied on a chapter-by-chapter basis. To assess the ability of each mined q matrix to describe the observed learning, the PFA model of Pavlik et al [4] is fitted to it and a cross-validated AUC is calculated. The models are assessed based on whether PFA's predictions of student correctness are accurate.

Early results show that both algorithms do a reasonable job of describing student progress, but q matrices with very different numbers of KCs fit observed data similarly well. Consequently, further consideration is required before automated extraction is practical in this context.

CCS Concepts

•Information systems → Data mining; •Applied computing → Computer-assisted instruction; Learning management systems; E-learning;

Keywords

Knowledge Components; Knowledge Tracing; Data Mining

1. INTRODUCTION

The LearnSmart platform¹ is an adaptive learning system that follows along with the textbook used in a course; stu-

¹<http://www.mheducation.com/highered/platforms/learnsmart.html>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

LAK '17 March 13-17, 2017, Vancouver, BC, Canada

© 2017 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-4870-6/17/03.

DOI: <http://dx.doi.org/10.1145/3027385.3029462>

dents are given questions corresponding to topics in each chapter in an order prescribed by the system, and must correctly answer a certain number of questions from each topic before the platform has deemed that they have mastered the chapter. LearnSmart is very widely used; in an average month, there are more than 100,000,000 distinct student-question interactions.

The actual learner experience in LearnSmart is dependent on metadata produced by subject matter experts, namely: what topics exist in each chapter, and how much evidence does a correct response to a given question contribute to proving that a student has mastered the topic? This paper discusses part of a broad effort to confirm the validity of this metadata, first by mining knowledge components from student interactions and then (as part of a future work), to trace the acquisition of these components and confirm that students are not provided with too little or too much practice on each topic.

This paper focuses on a single chapter from a single course teaching the Spanish language to English speakers. In this chapter there are 133 questions, given to roughly 11,000 students in a sequence deemed appropriate by the adaptive platform. In total, there are approximately 750,000 distinct student-question interactions in this dataset.

2. METHODOLOGY

In this work, we attempt to extract KCs from “chapters” automatically, as the topic and skill tagging is absent in some textbooks, and the sheer volume of texts makes manual tagging impractical. This approach assumes that KCs will not be found across chapters; while this assumption is unlikely to be formally true, the pedagogy of this platform is such that we want to measure KC progress on a chapter-by-chapter basis.

2.1 Imputing missing features

While students do not have any control over the sequence of questions that they seem the adaptive platform generally deems students to be finished with a given chapter after showing them roughly half the questions. While the algorithm of Barnes [1] can be easily adapted to work with missing data, regression techniques like non-negative matrix factorization (NMF) cannot.

Therefore, for each question, a logistic model is built using all students who did attempt the question as training data, where the features are the students' interactions on all other questions in the chapter, using one-hot encoding to turn

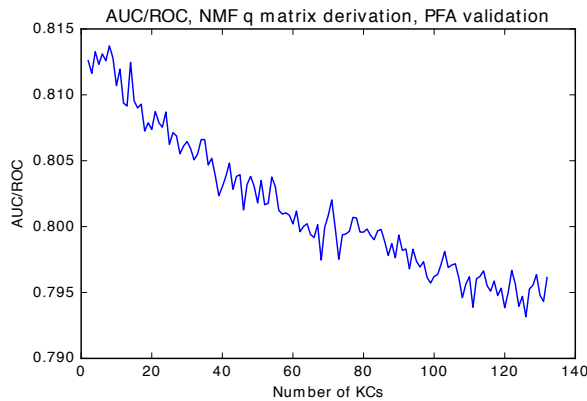


Figure 1: AUCROC of PFA, predicting student performance using q -matrices derived via NMF. Note the limited range of AUCROC values, shown in the Y axis.

{correct, incorrect, did not attempt} into binary features. A single regularization parameter is selected for all equations based on the cross-validated prediction accuracy among the training set, but the logistic imputation is used only to fill in missing data points; observations are never overwritten.

2.2 Extracting Knowledge Components

Knowledge components are customarily mapped to questions or items in a “ q -matrix”, a binary matrix which contains one column for each question and one row for each KC. [6] A custom implementation of the q -matrix algorithm outlined by Barnes [1] was implemented in Apache Spark. Concisely: random q -matrices of high sparsity containing n KCs are generated, and in random order each entry in the matrix is flipped from 0 to 1 or vice versa; if the new matrix is better able to describe actual student interactions the changes are kept. Unfortunately, this stochastic search is very slow, sufficiently so that for now it is set aside.

Using NMF techniques, as outlined by [3], produces results many hundred times faster. Using the NMF factorization package built into scikit-learn and Apache spark [5], it can find q -matrices spanning the range from two to 133 (i.e., the number of questions in the chapter, or the largest size at which matrix factorization would make sense) in a few hours on a modern desktop computer.

2.3 Scoring Knowledge Components

Finally, to score the descriptive ability of each q -matrix, an implementation of the PFA knowledge tracing model [4] was implemented in Apache Spark. Only the knowledge tracing part of the algorithm was implemented. The score of a q -matrix is judged to be the AUC/ROC of the PFA model predicting a correct answer for each student appearance, using a split testing and training set.

So far, this has not yielded conclusive results: while the q -matrices uncovered by NMF produce a fairly high AUC (≈ 0.8), it is found to be very insensitive to the number of KCs mined. The range of AUCs varies by less than 1% along the entire [2, 131] range of q . We believe that this is due to the dominance of the question-difficulty term in PFA.

3. CONCLUSIONS AND NEXT STEPS

There are two immediate and parallel paths that need to be pursued: ensuring that the flat AUC is not an artifact of either the NMF for factoring or PFA for scoring (e.g., by comparing with the algorithm in [1] or with Bayesian Knowledge Tracing [2], or by removing the difficulty component in PFA).

If this flat AUC is a robust feature of the data, then NMF should be revisited using matrix factorization technique that does not require imputing missing data, at least to confirm that it produces q matrices similar to those produced when imputing data. Additionally, comparing the 2-means method of turning non-binary matrices into binary matrices, used by Desmarais, with explicitly binary matrix factorization techniques (e.g. [7]), would help ensure that using NMF to produce binary matrices produces comparable results.

4. ACKNOWLEDGMENTS

The authors would like to thank the MHE LearnSmart team for providing access to data used in this analysis.

5. REFERENCES

- [1] T. Barnes. The q -matrix method: Mining student response data for knowledge. In *Proceedings of the AAAI-2005 Workshop on Educational Data Mining*, pages 39–46. Association for Advancement of Artificial Intelligence, Jul 2005.
- [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, 1994.
- [3] M. C. Desmarais, B. Beheshti, and R. Naceur. Item to skills mapping: Deriving a conjunctive q -matrix from data. In *International Conference on Intelligent Tutoring Systems*, pages 454–463. Springer Science & Business Media, 2012.
- [4] P. I. Pavlik, H. Cen, and K. R. Koedinger. Performance factors analysis – a new alternative to knowledge tracing. In *Proceedings of the 2009 Conference on Artificial Intelligence in Education: Building Learning Systems That Care: From Knowledge Representation to Affective Modelling*, pages 531–538, Amsterdam, The Netherlands, The Netherlands, 2009. IOS Press.
- [5] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [6] K. K. Tatsuoaka. Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of educational measurement*, 20(4):345–354, 1983.
- [7] Z. Zhang, T. Li, C. Ding, and X. Zhang. Binary matrix factorization with applications. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, pages 391–400. Institute of Electrical and Electronics Engineers, Oct 2007.