Curio: An On-Demand Help-Seeking System on iTextbooks for Accelerating Research on Educational Recommendation Algorithms

Ying-Jui Tseng[1], Yu-Hsin Lin[1], Gautam Yadav[1], Norman Bier[1] and Vincent Aleven[1]

[1]Carnegie Mellon University, 5000 Forbes Ave Pittsburgh PA 15213, United States

Abstract

The emergence of intelligent textbooks (iTextbooks) has broadened the landscape of education, yet poses new challenges in meeting the needs of diverse learners. In response to these challenges, we developed Curio, a personalized educational recommendation system for help-seeking embedded within iTextbooks. By using video transcripts and text extracted from images in iTextbooks as an index for a search engine, Curio provides targeted and context-specific content to aid comprehension. Integrated within the iTextbooks itself, this on-demand tool offers instant clarification on complex STEM concepts, making learning more adaptable to individual needs. We have been working with Open Learning Initiative (OLI) to refine and test Curio's potential, and envisage its application across various iTextbook platforms. This paper discusses Curio's contribution to the educational recommendation algorithms field.

Keywords

Self-paced Learning, Personalized E-learning, Intelligent Textbooks, Help-Seeking, Recommendation Systems

1. Introduction

The transition from traditional textbooks to intelligent textbooks (iTextbooks) has democratized access to education on an unprecedented scale [1]. However, with increased accessibility comes greater learner diversity among learners, with varying levels of prior knowledge and learning goals. This diversity poses a significant challenge to online learning. The current lack of personalized scaffolding for video learning, particularly in STEM fields, exacerbates the issue. STEM concepts are often interdependent, with formulas and code often sitting at the intersection between related concepts. For novice learners, these representations are a particular challenge, presenting what is often quite literally a new language that is exceptionally dense – formulas, for example, often succinctly encapsulate expert-level skills and knowledge. We focus on these representations because they often form real barriers; learners may struggle to continue their learning if they cannot grasp a single formula or line of code. The current

AIED'23: Symposium on the irreproducible science, June 07–11, 2021, Woodstock, NY Artificial Intelligence in Education Tokyo, Japan, July 7, 2023
yingjuit@andrew.cmu.edu (Y. Tseng); yuhsin.lin@outlook.com (Y. Lin); gyadav@andrew.cmu.edu (G. Yadav); nbier@andrew.cmu.edu (N. Bier); aleven@cs.cmu.edu (V. Aleven)
0009-0006-1801-6061 (Y. Tseng); 0000-0003-4384-7191 (Y. Lin); 0000-0002-9247-1920 (G. Yadav); 0000-0002-1581-6657 (V. Aleven)

help-seeking mechanisms, such as discussion forums, have low engagement levels due to a lack of timely feedback [2]. External search engines and video platforms such as YouTube are often used, but they can be unreliable and non-educational information in the search result produces extraneous cognitive load [3].

Curio seeks to address these challenges by recommending learning content within the learning platform and showing the results in the context of the video player. This approach simultaneously reduces the friction of leaving the platform while ensuring the quality of search results. Moreover, the tool can leverage the context of the question and the learner model to filter on-demand recommendations to individual learners' current needs, making learning experiences more personalized. By lowering barriers to help-seeking, Curio aims to not only improve domain-specific learning outcomes but also increase learners' self-efficacy.

2. Background

2.1. Help Seeking in Interactive Learning Environments

In interactive learning environments (ILEs), help-seeking is a pivotal strategy that promotes the development of independent skills and abilities [4, 5, 6]. Context-sensitive help content in ILEs may provide higher quality assistance than peer helpers . [7].

Help-seeking involves a series of processes, such as becoming aware of the need for help, deciding to seek help, identifying potential helpers, using strategies to elicit help, and evaluating the help-seeking episode [5, 6, 8]. Furthermore, placing help under the control of the learner is likely to improve the timing of the explanations, making them most useful for constructing new knowledge [9]. Help-seeking effort can be reduced by impacting variables that mediate the relations between prior knowledge and help-seeking. [7]. Adaptive ILEs that minimize demands on students' help-seeking skills can be beneficial, especially in iTextbooks that inherently demand these skills.

Existing help-seeking mechanisms in digital learning environments, such as discussion forums, have low levels of engagement [10]. In response, the majority of students resort to existing resources like books and search engines for problem-solving, thus suggesting the need for exploring alternative methods of providing support to learners, such as content recommendation [11].

2.2. Search in Learning Context

While search engines have been valuable tools in educational contexts, there is a pressing need to curtail the cognitive overload that learners experience from non-educational snippets in search engine result pages [3]. This can be achieved by only presenting educational content in our proposed system.

Furthermore, research is sparse on the optimization of extraction and ranking algorithms in search engines based on learning needs. No current algorithm is interested in determining the learner knowledge level from the request or the elimination of non-educational content from search engines.

2.3. Educational Recommendation Systems

Educational recommendation systems have received substantial attention over the years, with research investigating various aspects such as machine learning-based recommendation systems for e-learning [12], ontology-based recommender systems for e-learning [13], and recommender systems to support learners' agency in a learning context [14].

Educational recommendation systems that adapt to students' existing knowledge of specific domain concepts have demonstrated efficacy in delivering personalized scaffolding within iTextbooks [15]. For example, researchers have constructed a Wikipedia recommendation interface within the digital textbook system to provide learners with alternate educational resources. This system aids learners in gaining prerequisite knowledge when they struggle with a particular question [16].

2.4. Learner Modeling

Learner modeling is a key aspect of adaptive learning systems, encapsulating the systems' assumptions about a learner's unique attributes related to educational processes [17]. These models can account for various learner characteristics, such as demographic information, knowledge level, cognitive characteristics, social attributes, personality traits, and motivation factors [18, 19]. Recent literature reviews highlight the growing interest in this area and the diversity of techniques used for learner modeling, such as machine learning techniques, Bayesian Knowledge Tracing (BKT) [20], and other hybrid models.

For instance, the Skills Map learner model of OLI platform applies BKT to classify learners' knowledge into states of 'learned' or 'not learned'[21]. This method delineates probabilities for learners' responses premised on their comprehension of a concept[22].

Curio functions as an extensible tool capable of integrating learner models within iTextbook platforms to further personalize content recommendations for help-seeking. For example, we intend to incorporate the Skill Map model from the OLI platform to enhance the relevance of video recommendations based on a learner's prior knowledge. If a learner has successfully mastered a skill, Curio's recommendation algorithm will prioritize videos that do not pertain to this skill. Conversely, if the learner is in the process of understanding a skill, the system will prioritize videos relevant to that skill, thereby assisting the learner in grasping the concept.

3. User Interface

The Curio user interface is a comprehensive solution designed to enhance the e-learning experience by being implemented as an overlay atop any video player or text within various learning platforms. It has been structured to facilitate easy access to learning resources, promote interactivity, and provide tailored explanations for students' misconceptions.

3.1. In-Widget Recommendation Result

Curio's In-Widget Recommendation feature serves as a powerful tool that provides learners with a range of educational resources directly within the platform. The feature operates by

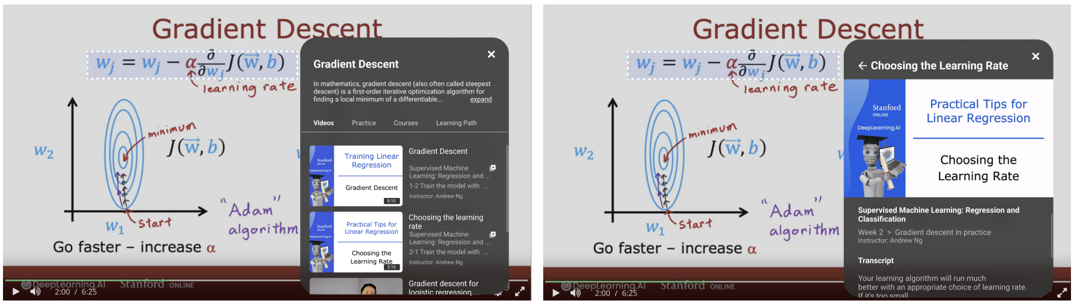

Figure 1: In-Widget Recommendation Result: The learner has selected an area within the video player corresponding to a challenging formula. In response, Curio generates a list of recommended videos and resources that directly tackle the concepts and formulas the learner is struggling with. Each recommended resource is displayed as a clickable thumbnail accompanied by a brief description of its content. By choosing a recommended video, the learner can view it within the widget itself, thereby preserving the continuity of their learning experience.

analyzing the content currently being viewed by the learner and generates relevant educational materials to supplement their learning. These results, organized according to their relevance to the current learning context, are directly presented to the learners within the video player (see Figure 1) or text interface, without necessitating a disruptive shift to a new page or platform.

3.2. Concept Description

Curio takes advantage of large language models (LLM) to supplement the recommendation results, specifically integrating GPT-4 to provide text-based instructions. This feature targets students' misconceptions, delivering tailored guidance to clarify their understanding and address gaps in their knowledge. We crafted a example result and explicly providde regulations to the GPT-4 endpoint, to ensure the quality of generated summaries. The main prompt body we used in the query is

What's the definition of *$text* related to *$learningObj* in a course *$courseName*?

By implementing the few-shot prompting technique and supplying the context of the learning material the learner is seeking help with as variables in the GPT-4 prompt, Curio is equipped to deliver context-specific, real-time explanations for any term, formula, and code snippets within iTextbooks.

3.3. Collection Feature

The Collection Feature in Curio is designed to facilitate personalized learning by allowing learners to earmark specific resources for future reference. This feature is particularly advantageous for learners who discover relevant materials that they wish to revisit later. This ability to curate learning resources based on individual needs further enhances the personalization capabilities of Curio.

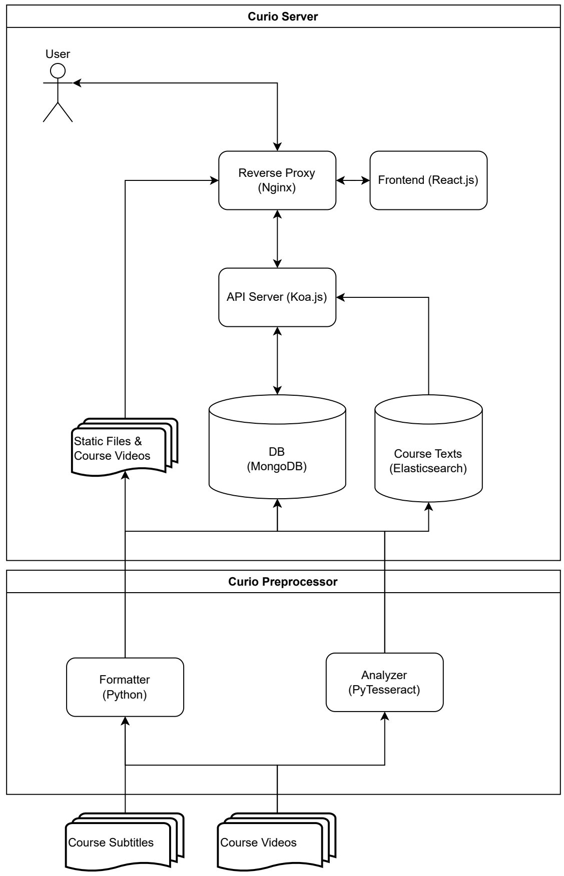

Figure 2: Curio's System Architecture

In summary, the Curio user interface serves as an embedded help-seeking tool within learning platforms. It provides on-demand, uninterrupted access to tailored educational resources, thus offering personalized scaffolding for learners according to their unique needs.

4. System Implementation

The primary objective of Curio's system implementation is to address three key technical challenges that may hinder users in their quest to comprehend learning contents on the platform. The first challenge is how to effectively index all teaching materials available on the platform. The second challenge is how to guide users in conducting effective searches with appropriate keywords. Lastly, the third challenge is how to ensure that when users query the platform, they do not receive excessive false positives.

To tackle the first challenge, we employed Tesseract v5.0 [23], an open-source LSTM-based OCR solution. This solution enables us to scan the video frame every few seconds and extract all text, source code, and formulae displayed on the screen. We store this information, along with subtitles and timecodes, in Elasticsearch [24], which facilitates full-text searching.

Additionally, we integrated the Tesseract engine into our video player with the same configurations, which enables students to initiate searches by using a selection tool on the interface and encourages them to search with the same vocabulary set stored in the Elasticsearch server.

Lastly, to mitigate the risk of false positives, we apply a weight calculated based on the learner's current proficiency on prerequisites, ensuring that users receive relevant learning materials. Currently, we are testing the platform with a simpler version of algorithm (L_{Obj} indicates whether a returned video share the same learning object with current video or not):

$$\text{newScore} = (W_1 \cdot (1 - L_{Obj}) + W_2 \cdot L_{Obj}) \cdot \text{originalScore}$$

$$W_1 = 0.5, W_2 = 1.5$$

While advanced learner models have not been integrated, Curio was intentionally designed as an extensible approach capable of accommodating such enhancements. This design allows for further improvements to the search results by incorporating advanced learner models if desired.

5. Future Work

Curio aims to accelerate learning science research by empowering researchers to experiment with different educational recommendation algorithms and collect valuable help-seeking data that is traditionally difficult to measure. Research on optimization of extraction and ranking algorithms based on learning needs is relatively rare, and existing algorithms do not aim to determine the learner's knowledge level from the request. Curio can address this issue by enabling researchers and developers to experiment with different educational recommendation algorithms on both established platforms and research prototypes, facilitating data collection. Researchers can import platform- or lesson-specific video metadata and learner models into Curio's Elasticsearch service and try out different weights to optimize recommendation strategies for specific research questions.

In addition to accelerating research in educational recommendation algorithms, Curio can help researchers and educators gain insights into learner's help-seeking behavior. One of the major challenges in measuring the help-seeking behavior of iTextbook learners is that it often happens on external platforms like Google search and YouTube. Curio offers a solution by recommending learning content within the iTextbook, enabling researchers to gain insights into learners' help-seeking behaviors and providing guidelines for optimal educational content recommendation strategies. Additionally, the help-seeking data collected by Curio can serve as a supplement to measure iTextbook learners' self-regulation, agency, and curiosity, which are nowadays measured by self-report questionnaires that raise concerns about susceptibility to bias.

The data collected by Curio will be openly published and uploaded to DataShop, an open-sourced repository for sharing and analyzing data on the interactions between students and educational software. Furthermore, researchers who utilize Curio for research can also collect and share the help-seeking data they collect with our data pipeline and infrastructure.

References

[1] P. Brusilovsky, S. Sosnovsky, K. Thaker, The return of intelligent textbooks, AI Magazine 43 (2022) 337–340.

[2] L. Corrin, P. G. De Barba, A. Bakharia, Using learning analytics to explore help-seeking learner profiles in moocs, in: Proceedings of the seventh international learning analytics & knowledge conference, 2017, pp. 424–428.

[3] J. K. Homte, B. Batchakui, R. Nkambou, Search engines in learning contexts: A literature review, International Journal of Emerging Technologies in Learning (iJET) 17 (2022) 254–272.

[4] R. Ames, Help-seeking and achievement orientation: Perspectives from attribution theory, New directions in helping 2 (1983) 165–186.

[5] S. Nelson-Le Gall, Help-seeking: An understudied problem-solving skill in children, Developmental review 1 (1981) 224–246.

[6] R. S. Newman, Adaptive help seeking: A strategy of self-regulated learning, Self-regulation of learning and performance: Issues and educational applications (1994) 283–301.

[7] V. Aleven, E. Stahl, S. Schworm, F. Fischer, R. Wallace, Help seeking and help design in interactive learning environments, Review of educational research 73 (2003) 277–320.

[8] M. Puustinen, Help-seeking behavior in a problem-solving situation: Development of self-regulation, European Journal of Psychology of education 13 (1998) 271–282.

[9] A. Renkl, Worked-out examples: Instructional explanations support learning by self-explanations, Learning and instruction 12 (2002) 529–556.

[10] D. Onah, J. Sinclair, Assessing self-regulation of learning dimensions in a stand-alone mooc platform (2017).

[11] D. Davis, G. Chen, C. Hauff, G.-J. Houben, Activating learning at scale: A review of innovations in online learning strategies, Computers & Education 125 (2018) 327–344.

[12] A. Klašnja-Milićević, B. Vesin, M. Ivanović, Z. Budimac, L. C. Jain, A. Klašnja-Milićević, B. Vesin, M. Ivanović, Z. Budimac, L. C. Jain, Recommender systems in e-learning environments, E-learning systems: Intelligent techniques for personalization (2017) 51–75.

[13] H. Drachsler, K. Verbert, O. C. Santos, N. Manouselis, Panorama of recommender systems to support learning, Recommender systems handbook (2015) 421–451.

[14] J. L. Santos, S. Govaerts, K. Verbert, E. Duval, Goal-oriented visualizations of activity tracking: a case study with engineering students, in: Proceedings of the 2nd international conference on learning analytics and knowledge, 2012, pp. 143–152.

[15] K. Thaker, L. Zhang, D. He, P. Brusilovsky, Recommending remedial readings using student knowledge state., International Educational Data Mining Society (2020).

[16] B. Rahdari, P. Brusilovsky, K. Thaker, J. Barria-Pineda, Knowledge-driven wikipedia article recommendation for electronic textbooks, in: Addressing Global Challenges and Quality Education: 15th European Conference on Technology Enhanced Learning, EC-TEL 2020, Heidelberg, Germany, September 14–18, 2020, Proceedings 15, Springer, 2020, pp. 363–368.

[17] A. Abyaa, M. Khalidi Idrissi, S. Bennani, Learner modelling: systematic review of the literature from the last 5 years, Educational Technology Research and Development 67 (2019) 1105–1143.

[18] R. Pelánek, Bayesian knowledge tracing, logistic models, and beyond: an overview of

learner modeling techniques, User Modeling and User-Adapted Interaction 27 (2017) 313–350.
[19] K. Seta, Y. Taniguchi, M. Ikeda, Learner modeling to capture meta-cognitive activities through presentation design, the Journal of Information and Systems in Education 14 (2015) 1–12.
[20] S. Bulathwela, M. Pérez-Ortiz, E. Yilmaz, J. Shawe-Taylor, Semantic truelearn: Using semantic knowledge graphs in recommendation systems, arXiv preprint arXiv:2112.04368 (2021).
[21] N. Bier, S. Lip, R. Strader, C. Thille, D. Zimmaro, An approach to knowledge component/skill modeling in online courses, Open Learning (2014) 1–14.
[22] M. V. Yudelson, K. R. Koedinger, G. J. Gordon, Individualized bayesian knowledge tracing models, in: Artificial Intelligence in Education: 16th International Conference, AIED 2013, Memphis, TN, USA, July 9-13, 2013. Proceedings 16, Springer, 2013, pp. 171–180.
[23] GitHub - tesseract-ocr/tesseract: Tesseract Open Source OCR Engine (main repository) — github.com, https://github.com/tesseract-ocr/tesseract, 2023. [Accessed 20-May-2023].
[24] Elastic, Elastic/elasticsearch: Free and open, distributed, restful search engine, 2023. URL: https://github.com/elastic/elasticsearch, [Accessed 20-May-2023].