CollectiveTeach: Crowdsourcing Lesson Plans

Ashwin Venkataraman
Dept. of Computer Science
NYU Courant
ashwin@cs.nyu.edu

Rishabh Ranawat
New York University
rdr324@nyu.edu

Sepehr Vakil
Dept. of Curriculum and Instruction
College of Education, UT Austin
svakil@utexas.edu

Jay Chen
Dept. of Computer Science
NYU Abu Dhabi
jchen@cs.nyu.edu

Srikanth Jagabathula
IOMS Dept.
NYU Stern Business School
sjagabat@stern.nyu.edu

Lakshminarayanan Subramanian
Dept. of Computer Science
NYU Courant
lakshmi@cs.nyu.edu

ABSTRACT

Lack of quality textbooks and good educational resources is a well-known problem in developing regions. In this paper, we describe the design of CollectiveTeach, a web platform that aims to integrate rich educational content into an inquiry-based framework, viz. the 5E learning model, for generating web-annotated lesson plans for school and college teachers in developing countries. Given the wealth of educational resources on the Web, CollectiveTeach helps teachers to author new lesson plans through a simple web interface allowing them to easily search, select, order and collect educational content relevant to the topics they wish to teach. This paper describes our experiences building two versions of the CollectiveTeach platform. The initial platform was tested via a user study with a cohort of 19 teachers in Ghana and the learnings of this user study were incorporated in a second, improved version of CollectiveTeach; the current prototype of CollectiveTeach was evaluated using human experts for computer science subjects covered in standard undergraduate curriculum.

1 INTRODUCTION

Over the last several decades, there has been great progress in integrating inquiry-based teaching strategies into the classroom [15, 16]. Simultaneously, there has been an explosion of online web-based learning materials and initiatives to introduce computers into classrooms [9, 25]. Unfortunately, teachers have only a few tools for making productive use of online content within inquiry-based frameworks for teaching and learning. This problem is worse in developing regions where lower levels of technological literacy present severe challenges to teachers seeking to incorporate web resources in their teaching materials and classroom instruction.

This paper presents the design, implementation and early deployment experiences of CollectiveTeach, an online lesson plan generation platform that enables teachers to collectively organize web-based educational resources within an inquiry-based framework for teaching and learning. The design of CollectiveTeach draws inspiration from the 5E model [10] based on the educational philosophy and psychology of Johann Herbart [23], which has a long history in educational theory grounded in ideas of Piaget and Dewey [29]. The 5E model for preparing a lesson plan comprises of 5 stages: Engage, Explore, Explain, Elaborate and Evaluate. CollectiveTeach enables teachers to express the key topics in their lesson plan across these 5 stages in the 5E model and translates the teacher specifications to a list of appropriate and *targeted (web-) search queries*. CollectiveTeach presents a simple interface for teachers to inspect the top-ranked search results in each stage to select and order relevant web content for each part of their lesson plan; collectively, the user-chosen content coupled with the corresponding URLs forms a web-annotated lesson plan created by a teacher for a specific educational class. In addition, CollectiveTeach supports *search* and *upload* capabilities that enable teachers to incorporate multimedia rich content including video, images, presentations, documents and domain-specific educational resources without being overly complex in terms of user interface elements or additional features.

We present our experiences developing two versions of the CollectiveTeach platform across a multi-year effort. In 2013, we developed an early prototype version of the CollectiveTeach platform and evaluated the effectiveness of the platform based on interactions with 19 K-12 mathematics and science teachers in Ghana during two week-long workshops. We observed that the 5E model naturally fit the existing instruction style of most teachers who participated in our user study. Results from the study show that teachers broadly found CollectiveTeach to be a highly effective tool for both creating inquiry-based lesson plans and also integrating web-based multimedia content in mathematics and science classes. Despite the fact that the teachers rarely used web content when creating existing lesson plans and most of them were only formally introduced to the 5E model in our training sessions, we found that the teachers were able to quickly and easily use our platform.

The second version of the CollectiveTeach platform addressed many of the usability and content organization challenges that were present in the first version. The current prototype version supports a much richer set of functionalities and enables easy creation of lesson plans for new subjects within a few minutes. We evaluate the effectiveness of the second prototype version of CollectiveTeach

Workshop on Advancing Education with Data, KDD'17, August 14, 2017, Halifax, NS, Canada.

DOI: 10.1145/nnnnnnn.nnnnnnn

using a human expert-based rating approach for popular computer science subjects to determine the utility and coverage of content shown by the generated lesson plans. In summary, we believe that CollectiveTeach can be adopted as an effective platform for enabling easy creation of web-based lesson plans using online educational resources.

2 RELATED WORK

A growing community of scholars including educational researchers and computer scientists are interested in how technology can support new modes of learning and instruction. Research on the design of learning environments that leverage the rapid advancement of information technologies can be found in programs such as the Web-based Inquiry Science Environment (WISE) [4] and Learning Technologies in Urban Schools (LeTUS) among others [2]. Many of these projects explore the affordances of specific technologies such as visualizations, models, and data probes for inquiry-based learning. In addition to student supports for learning, the WISE program mentioned above also provides teacher supports for authorship and customization of curriculum [31, 32]. A product of over 20 years of research on learning with educational technologies, WISE is an example of a powerful educational tool based on traditions within constructivist philosophy, in particular the Knowledge Integration framework. The design of the CollectiveTeach platform was strongly informed by projects such as WISE, in particular the approach of incorporating constructivist perspectives on learning within the design of the platform itself. However, a limitation of programs such as WISE within the developing region context is complexity. The WISE tool includes advanced features that may pose significant challenges for teachers (and students) who have limited experience with computers and learning technologies [28].

In conjunction with the growth of the Web, there has been a similar growth in number of projects that have leveraged technology and web-based resources for education [2, 3, 19], especially for developing regions [1, 6, 11, 17, 19, 22, 30, 34]. The Hole-in-the-Wall project showed that exposure of web-based educational resources to under-privileged students without any formal guidance on innovative learning platforms can indeed produce positive learning outcomes [30]. Digital Study Hall is another successful project that has used participatory videos to enhance rural school education [34]. There have also been studies which have analyzed the merits of Powerpoint slides [27] and video materials [3, 21] as teaching aids. Recent work looking at textbook content in developing regions expose the limitations of existing educational resources [14, 20] and propose methods to automatically augment existing course materials with online content [6]. The basic rationale behind these works is similar to the focus of our paper in that the web has a wealth of educational information that can be used for enhancing classroom content. Simple tools and interfaces can dramatically lower the barriers to entry [7]. Chakraborty et al. found that there was sufficient quality content on the web and that many web resources were ideal for activity-based learning [11]. The major drawbacks of such systems is that despite the large amount of educational content, finding the most relevant content for a particular class may take a long time. Therefore, filtering the appropriate content and helping to organize the content for instructors is a central problem that is addressed in the design of CollectiveTeach .

3 THE COLLECTIVETEACH PHILOSOPHY

The basic building block of CollectiveTeach is the concept of an *atomic learning unit (ALU)*, which is a concise and compact educational material (spanning less than a few pages of textbook content) that primarily discuss a single concept. This is similar to the concept of "key section" introduced by Agrawal [5]. Given our focus on web based educational resources in the CollectiveTeach framework, an atomic learning unit can refer to the content in a single web page, relevant pages within an online document, or user posted content within the CollectiveTeach system. An atomic learning unit can refer to different types of content that pertain to the concept such as: (a) textual content that explains the relevant concept; (b) a collection of problems and solutions; (c) multimedia content (video, images); (d) user-uploaded content (pdfs, powerpoints, etc.). In the CollectiveTeach framework, an ALU refers to a single educational resource link on the Web or a user-uploaded file that provides concise educational material relating to a given concept.

Consider a lesson to represent the material covered in a single lecture in a traditional educational setting. We consider a lesson to be a collection of related concepts. In the CollectiveTeach framework, a *lesson plan* is an ordered collection of atomic learning units that is necessary to teach the concepts in a lesson. ALUs serve as a logical partition of the instructional content of a lesson with the purpose of effectively communicating concepts to students.

The CollectiveTeach lesson plan creation framework is directly inspired by the 5E model in the education pedagogy literature [24]. Building off prior learning cycle models, the Biological Sciences Curriculum Study (BSCS) 5E model is based on educational research tracing back to the early constructivist perspectives of Piaget and Dewey. As a robust philosophical tradition within education, constructivism is fundamentally concerned with how knowledge is constructed and therefore has significant implications for theories of instruction and curriculum development. The 5E model is one of the best known within this tradition. The BSCS 5E model contributes two additional stages, engagement and evaluation, to the Science Curriculum Improvement Study (SCIS) learning cycle which was comprised of three stages: exploration, invention, and discovery [10]. Not intended as a linear formula but rather a guide to structure activity in a way that centers students in the learning process, the 5E learning cycle consists of five stages: *Engage, Explore, Explain, Elaborate,* and *Evaluate.*

Engage: The first stage in the learning requires engaging the students. Engagement generally involves connecting the topic of instruction to the lived experiences of students through culturally relevant pedagogical practices. This may include showing videos, open-ended discussions, freewrites, or class debates designed to "hook" students into the topic of instruction [10].

Explore: Next, through hands-on activities, labs, or class discussions, students are guided to explore a topic that may have emerged in the engagement phase, or may relate more centrally to the topic of instruction. Explore activities generally help the teachers gain a sense of student prior knowledge, which skillful teachers will take into account as they unfold the rest of their lesson.

Explain: The new ideas generated, and the questions raised during exploration activities will help teachers target their instruction during the explain stage. Often during this step in the learning cycle, teachers will confirm student ideas or help clarify student misconceptions revealed during earlier stages in the process. Concrete activities usually involve direct instruction and presentation of scientific terms or ideas.

Elaborate: The elaboration stage is intended for deeper inquiry into the topic of instruction by challenging students with complex problems and demonstrating real-world applications of the concepts discussed. Common activities may include group problem-solving challenges and group discussions connecting topic of instruction to real-world applications.

Evaluate: Finally, the evaluation stage provides an opportunity for teacher and students to assess understanding and conceptual mastery of the information provided throughout the course of the class.

4 COLLECTIVETEACH 1.0

CollectiveTeach 1.0 was designed as a simple prototype system that integrated easily into the existing educational ecosystem with minimal training, cost or maintenance. Our goal was to introduce a platform that would speak to the needs of our target population (teachers) using a collaborative design approach [8]. The CollectiveTeach 1.0 platform enables a teacher who intends to teach a class on a given topic to easily find relevant online educational resources and create a Web-annotated lesson plan for her class. The design of CollectiveTeach 1.0 assumes that the teacher has a rough flow of the list of topics she intends to cover in her class and the *lesson plan*, in essence, is a set of curated web contents carefully chosen (by the teacher) to best fit the material covered in class. For example, a Biology teacher teaching human anatomy could significantly benefit from a wealth of images/videos on the web describing anatomy of the human body as an educational resource for both teaching her class more effectively as well as an additional learning guide for her students. There are two significant research challenges that need to be addressed to design a system that meets the above objectives: (1) *Given the large volume of educational information on the web, how can teachers identify relevant and high quality materials that will support student learning in the classroom?* Embedded within this challenge is the task of formulating the right type of search query that will return relevant content among the top few search results. (2) *Given a rough outline of how a teacher may want to organize her materials for a class, how can she assimilate all the relevant web information she finds into a coherent lesson plan?*

To address both these challenges, CollectiveTeach 1.0 leverages the 5E instructional model and uses the following multi-step approach to enable teachers to create their lesson plans. First, CollectiveTeach 1.0 provides a simple HTML form interface that guides teachers to organize their lesson plan outline within the framework of the 5E model. For each stage in the 5E model, it prompts the teacher to describe the instructional activities they intend for students to interact with in that stage. Additionally, teachers are given the option to "add web resources" to enrich or support the specified activities, by choosing the type of resource (web, video, image, etc.) as well as a subset of common keywords that can enhance the search query and increase the likelihood of returning web results that are both relevant and of high quality. In our preliminary tests we found that for specific stages in the 5E model, adding special common keywords like "problems", "questions", "examples", "applications" etc. can significantly enhance the quality of the search results. After teachers have completed outlining their lesson plan across the 5E stages, CollectiveTeach 1.0 converts the user input into a set of appropriate search queries and returns the top 3 search results for each "resource" requested, in a common web interface front-end. Finally, our platform provides a simple interface to enable teachers to sift through the collection of all search results and select which results have the relevant content they are looking for. For multimedia- rich content, CollectiveTeach 1.0 provides a direct snapshot of the content to enable faster selection from the user. Since the result pages are organized across the different stages, the chosen collection of search results form the web-annotated lesson plan constructed by the teacher.

4.1 Early User Study

We tested the effectiveness of CollectiveTeach 1.0 with mathematics and science teachers in Accra, Ghana. We recruited 19 teachers from a school district in the Accra region comprising primary, middle school and high schools, and both private as well as government schools. The participants had between 6 and 29 years (mean of 14.8) of teaching experience. Five participants were female (26%). Fifteen participants had a computer at home (79%), and 10 had Internet at home (52%). Teachers were invited to participate in two workshops designed to explore the role of technology in education with a focus on math and science subjects.

In the first workshop, participants completed a 20-minute pre-survey designed to better understand their current computer and technology skills along with their approach to creating lesson-plans. In addition, a 30-minute orientation to the 5E model and tutorial of the CollectiveTeach 1.0 platform was conducted. Then, the participants used the CollectiveTeach 1.0 platform for 45 minutes to create web-based lesson plans and finally, completed a post-survey focused on gaining feedback on specific features of the platform. Approximately a week after the first workshop, 10 participants were invited again based on the quality of lesson plans that they created to participate in the second workshop which consisted of the following 3 steps:

(1) Participants created two new lesson plans using the platform based on their existing curriculum materials
(2) Participants assessed their own lesson plans on a Likert scale rating system across 4 dimensions:
    (a) Consistency of lesson plan with 5E instructional framework
    (b) Skillful integration of web-content into lesson plan
    (c) Search results match lesson plan
    (d) Overall rating of lesson plan
(3) Participants rated the other participants' lesson plans across the same dimensions

The process of participants assessing their own lesson plans, as well as lesson plans of other participants informs our analysis of the success of the CollectiveTeach 1.0 platform presented below.

Survey Results. After participants completed workshop #1 of the user study, they were given a post-survey. Table 1 summarizes these results. These figures indicate that overall participants found that the 5E model was helpful, the platform was easy to use, web

Table 1: Post-survey results (Workshop #1). (Likert scale 1=Strongly Disagree to 5=Strongly Agree)

I usually use web resources for lesson planning	2.7
The 5E model was helpful	4.6
I thought the platform was easy to use	4.1
The platform returned useful search results	4.1
Web resources were useful	4.0
Overall, the platform was helpful for the tasks	3.9
I would use this platform again	4.5

resources returned were useful, and the platform was helpful for the lesson plan construction process.

Usefulness of the Platform. During the second workshop, after participants constructed two lesson plans based on their own curriculum materials, they were asked to assess their lesson plans across several dimensions relating to overall quality. Each of the ten participants who participated in the second workshop created two new plans, resulting in 20 self-rated lesson plans. Of these 20 self-assessments, 7 were marked "strongly agree" and 13 "agree" when presented the following statement:

"I believe this lesson plan is created according to the principles of 5E, resulting in engaging activities that emphasize student-centered learning and conceptual mastery of the material."

When participants rated each other's lesson plans using the same prompt, the results were nearly identical, with 6 marks for "strongly agree", 12 for "agree", 1 for "disagree" and one left unscored. Overall, these results indicate that participants expressed confidence that the CollectiveTeach 1.0 platform assisted the process of mapping their existing curricular materials into a cohesive lesson plan, within the inquiry-based 5E framework. This matches closely with the results from Workshop #1 that indicated our participants found the platform useful for completion of lesson plan creation tasks.

4.2 Limitations of CollectiveTeach 1.0

Despite overall enthusiasm, teachers did express some skepticism that inquiry-based approaches could take root in the current educational climate. Participant critiques of the platform from the surveys and focus groups were collected and organized into three themes: technical, structural, and cultural. Participants expressed the following concerns about the CollectiveTeach 1.0 platform:

- *Not being able to go back to previous stages and edit the lesson plan, especially once the results were returned as participants had new ideas for their plans but were not able to add.* (technical)
- *Results from search not always useful, especially for the Ghanaian context. Some resources are quite foreign to our students. No local context.* (cultural)
- *Adoption of this tool not feasible without an increase in teacher knowledge.* (structural)
- *To be able to adopt this model here will require massive refurbishment of our computer labs* (structural)

5 COLLECTIVETEACH 2.0

Based on the limitations identified above, we enhanced CollectiveTeach 2.0 with the following five features—(1) **Query formulation**, (2) **Reordering, editing and searching**, (3) **Automated filtering**, (4) **Summarization** and (5) **Design and Presentation**. We outline these features in this section.

5.1 Generate Lesson Plan

The first component of the platform gives teachers the ability to generate lesson plans by specifying a minimal set of keywords that describe the contents that she wishes to cover in the respective lesson.

The teacher is presented with a simple HTML form which serves as the planning page for generating the lesson plan and consists of descriptive metadata, viz. Subject Name, Course title, Lesson title and a few keywords describing the lesson outline. The teacher fills in these details which initiates the lesson plan generation process; see Figure 1 for an example.

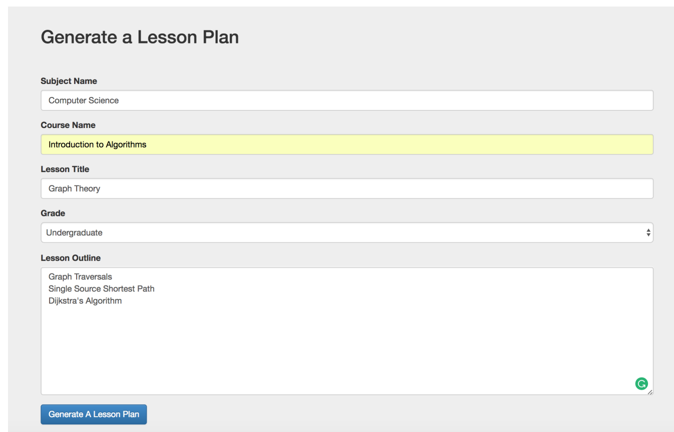

Figure 1: Generate Lesson Plan Interface

5.1.1 *Query Formulation.*

The addition of this feature to CollectiveTeach 2.0 increases the likelihood of improved search results. Based on the teacher-provided keywords, the platform formulates specific kinds of queries that aim to generate desirable and good-quality results. Specifically the teacher is not required to come up with the most effective terms for improving the search results, which can be hard in practice [26]. The query formulation adds three basic forms of structure to the search queries: (a) learning a set of domains relevant to a particular subject and constraining the search to pages within these domains; (b) targeting specific types of files that provide highly relevant content for given subjects; (c) adding specific key words that improve relevance and quality of search results.

As an illustrative example, to fetch content from trusted sources on the Web, we used search query formulation techniques such as appending "site:wikipedia.org" and "site:.edu" to the search queries, to help navigate the large space of educational content. Further, we also added targeted results for filetypes like PDFs and PPTs

(Powerpoint) by appending the query with "filetype:pdf" and "filetype:ppt" respectively. We targeted the results to `Wikipedia` pages for the early stages in the 5E model, since they usually contain comprehensive description and explanation of a particular concept (and also its related concepts). Further, utilizing the knowledge that professors around the world publicly post their course/lecture notes and conceptual explanations, we append terms such as 'concepts', 'notes', etc. to further improve the results. Such queries tend to return both conceptual as well as application-based results. Similarly, for the Evaluate phase the query is modified with terms like 'homeworks', 'exams', 'midterm', etc. which could return problem sets and exam papers from online educational websites. In general, the query formulation logic can be pre-encoded as a function of the subject using a small set of rules written by a user.

5.1.2 Automated Filtering. Another addition to the CollectiveTeach 2.0 platform is an automated filtering step that aims to discard irrelevant web links by applying empirically learnt heuristics. For instance, our lesson plan generator ignores links that only contain the course syllabus/outlines or course catalogs/schedules, as the mentioned resources do not provide any "instructional content". Other filtering techniques based on the content of the webpages can be naturally incorporated.

5.1.3 Presentation of Generated Results. To prevent user fatigue and/or overload, we aggregate the 5E model presented earlier into only two stages, which we refer to as Engage and Evaluate. This is representative of college (or university) education where typically for each course, there is a lecture each week that introduces and explains some concepts, followed by some form of evaluation like problem sets, homeworks, assignments, etc. We also display the top-level domain for each search result (like wikipedia.org or mit.edu) as well as the filetype (HTML, PDF, PPT etc.) to further enhance the presentation. See Figure 2 for an example.

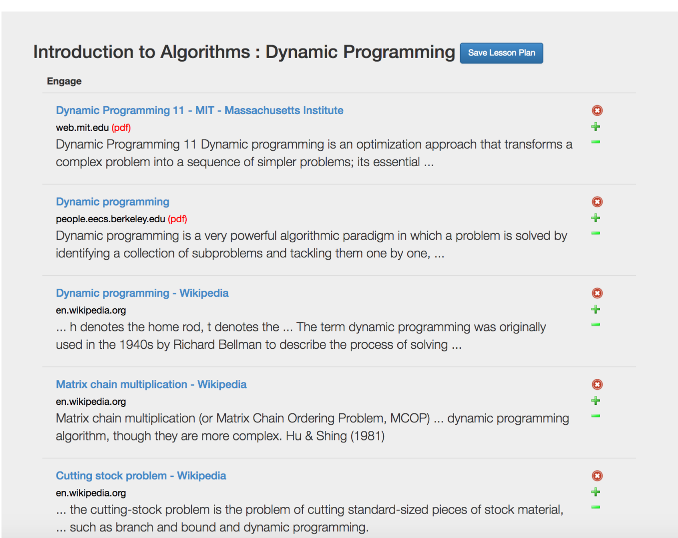

Figure 2: Lesson Plan for Dynamic Programming

5.1.4 Summarization of Generated Results. This feature aims at displaying a short summary of the webpage presented as part of the search result to help the teacher quickly gauge its relevance for incorporating it as part of her lesson plan, as opposed to going through the entire content of the webpage. Specifically, we use the Summarization Search paradigm introduced by Chakraborty et al. [12] that performs a detailed text analysis of any search result page to prepare a condensed summary. The summarization involves identifying portions of the webpages that have high relevance (or similarity) with respect to the terms in the search query.

5.1.5 Reordering and editing. The previous version of the platform did not allow teachers to go back and edit their lesson plans or reorder content within an existing lesson plan to match their presentation style. CollectiveTeach 2.0 provides both of these functionalities. Refer to Figure 2 which shows buttons at the extreme right of each search result that allow teachers to edit the generated lesson plans. After this selection process, the teacher can press the Save lesson plan button to add the lesson plan to her profile where she can view all the lesson plans that she has created.

5.2 Upload Lesson Plan

CollectiveTeach 2.0 allows teachers to create their own lesson plans by uploading content in the form of documents, images/videos or other multimedia content. This helps in alleviating some of the concerns raised by teachers in the user study earlier about web resources not providing the right local or cultural context. Further, teachers can also augment lesson plans generated using search results with their own (uploaded) content to obtain a lesson plan that most suits their need.

5.3 Search Lesson Plan

A teacher can also search for existing lesson plans in the system made by other teachers and potentially incorporate relevant resources into her own lesson plans; see Figures 3 and 4. Note that

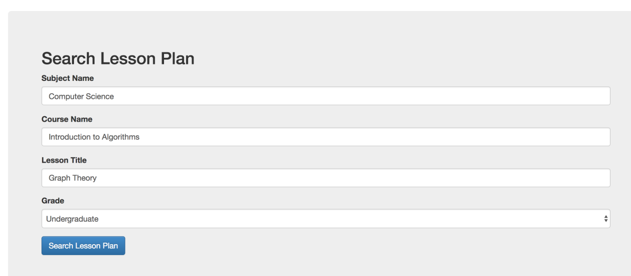

Figure 3: Search Lesson Plan Interface

this not only includes URLs but also documents and other educational material uploaded by teachers which might appear useful for improving the quality of the lesson plan. This feature of CollectiveTeach 2.0 allows teachers to share and re-use educational content thereby promoting collaboration and resulting in a "collective teaching" environment.

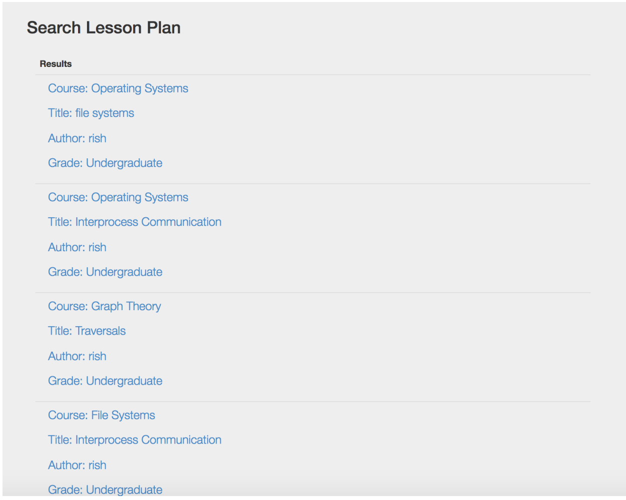

Figure 4: Example Lesson plans returned via Search

5.4 Extensions

In addition to the proposed features and capabilities above, CollectiveTeach 2.0 supports specific extensions including: a rating mechanism, offline mode of operation and multilingual support. Since our eventual goal is to improve education in developing regions, the platform should be able to operate in areas having poor (or zero) Internet connectivity as well as satisfy the needs of Non-English speaking communities.

5.4.1 Rating Mechanism. In order to improve the quality of content on the platform, we implement a voting mechanism for lesson plans. Users of the platform can rate (upvote/downvote) lesson plans in the system and the collective feedback from the "crowd" is used to rank the different lesson plans and improve the quality of search results. Specifically, the collected votes act as a ranking filter in the *Search Lesson Plan* phase of the platform (see above). This provides an organic way to maintain the content quality on the platform, without having to rely on dedicated curators to manage the content.

5.4.2 Offline Content. To enable the platform to operate without Internet connectivity, we create a local searchable index on a corpus of educational resources suited for a given region/subject. In particular, we adopt the following steps:

(1) **Crawler:** We built a multi-threaded crawler that is run periodically to download and index relevant educational content from the Web. In order to ensure quality and maintain diversity, the crawler has been tuned to download content from different `.edu` domains and `Wikipedia`. In the future, we aim to augment this list with additional websites.
(2) **Indexing and Search:** Search works in conjunction with query formulation and automated filtering described previously in Section 5.1. This extended version of the platform uses Elasticsearch[1] to index, rank and search over the offline educational resources.
(3) **Fallback to online capability:** In scenarios where the user is not satisfied with the offline content returned by Elastisearch, we can fetch additional content from the Web (assuming connectivity) as outlined above.

5.4.3 Multilingual Capability. Currently, the platform can support Spanish content (with English being the default). However, we can easily extend the *Generate Lesson Plan* phase to fetch content in a specific language (by modifying the parameters to the search engine API). Similarly, the crawler will now need to index domains belonging to the provided language; we are currently working on this.

6 ASSESSING COLLECTIVETEACH 2.0

In order to evaluate our platform, we carry out both quantitative and qualitative analysis of the lesson plans generated, using human experts. Specifically, we consider three Computer science courses that are part of the Undergraduate curriculum in most colleges and universities—`Algorithms`, `Operating Systems` and `Machine Learning`. Within the Algorithms course, we generate 10 lesson plans using the CollectiveTeach 2.0 platform where the lesson title and outline keywords are taken from 10 chapters of the popular CLRS [13] textbook. Similarly, we create 4 lesson plans for Operating Systems using the Tanenbaum [33] textbook and 4 Machine Learning lesson plans from an advanced undergraduate class.

We measure the utility of the generated lesson plans in two stages: (1) Stage 1 evaluates the fraction of search results shown by the platform that are relevant to the lesson plan; and (2) Stage 2 which measures the *usefulness* of the content presented for teaching the concepts desired, as well as its *coverage* in terms of the keywords provided by the teacher. We discuss the results of each stage next.

6.1 Stage 1: Analysis and Observations

For a given lesson plan, the human expert rated each search result as being relevant or not, and computed the fraction of results that were relevant for both the Engage and Evaluate phase separately. Then for each of the three courses, we computed the mean relevance across all the lesson plans. Figure 5 reports the results.

Course	Fraction of Relevant Engage Links	Fraction of Relevant Evaluate Links
Algorithms	0.75 ± 0.06	0.56 ± 0.08
Operating Systems	0.80 ± 0.05	0.47 ± 0.13
Machine Learning	0.89 ± 0.04	0.33 ± 0.12

Figure 5: Fraction of relevant search results

Algorithms. We used lesson titles such as "Dynamic Programming", "Divide & Conquer" and lesson outlines with keywords like "Rod cutting; Matrix-chain multiplication; Longest common subsequence". The fact that over 75% of the Engage results were relevant signals that for popular courses, our platform is able to fetch relevant lesson plans to a good extent. For instance, for the Dynamic Programming lesson plan, we obtain content from reputed universities like MIT, Berkeley, Rutgers, etc. as well as informative

[1] https://www.elastic.co/products/elasticsearch

Wikipedia pages. We observe that for the Evaluate stage, about 56% of the results were relevant.

Operating Systems: In this case, we used lesson titles such as "Deadlocks", "Input/Output" and lesson outlines with keywords such as "Resource Allocation Graphs; Banker's Algorithms; Safe and Unsafe states". Again about 80% relevance for the Engage phase shows that our system is able to formulate accurate queries that filter out the best results for the users. However, we obtain only 47% relevance for the Evaluate stage because our platform fetched midterm/final exams, as opposed to programming assignments which are more suited for the Operating Systems course.

Machine Learning: Here, we used lesson titles such as "Perceptron Classifier", "Stochastic Gradient Descent" and lesson outlines with keywords like "Gradient Descent; Subgradients; Convex-Lipschitz Bounded Function". An 89% relevance for an advanced undergraduate course is pretty high, showing that even simple query formulation techniques can fetch highly relevant results. One thing to note is that there were lesser number of links in general collected for the Machine Learning course compared to Operating Systems or Algorithms and thus, the "fractional" relevance could still be high, however, it would not be completely fair to compare the results across the different courses. One can still make the case, though, that even for advanced undergraduate courses, the platform is able to generate highly relevant Engage links. The relevance of the Evaluate links is much worse despite the plethora of ML related content available on the Web, which we aim to address in future work.

6.2 Stage 2: Analysis and Observations

The second stage of analysis involves another human expert evaluating the usefulness and coverage of the lesson plans that were generated by the human expert in Stage 1 of the analysis above. Specifically, all irrelevant links were discarded after the above analysis and the remaining collection of links were saved as lesson plans. The new human expert then rated the lesson plans on the two dimensions introduced above on a scale of 1 to 5, with 1 representing "Low" and 5 representing "High". Similar to the analysis above, we compute the average rating across the lesson plans in each course and the results are summarized in Figure 6.

	Engage		Evaluate	
Course	Usefulness	Coverage	Usefulness	Coverage
Operating Systems	4.75 ± 0.25	4.75 ± 0.25	2.75 ± 0.75	3.75 ± 0.75
Algorithms	3.90 ± 0.23	4.50 ± 0.23	3.1 ± 0.26	3.67 ± 0.33
Machine Learning	4.50 ± 0.29	4.25 ± 0.48	4.00 ± 1.00	4.5 ± 0.5

Figure 6: Usefulness and Coverage of the generated lesson plans

Algorithms: For the Engage phase we observe that the user found the lesson plans to be useful and to cover a good range of sub-topics for the different lessons. Also, these values seem to align with the relevance values from Stage 1 of the evaluation, where we had over 75% relevance rate, displaying a strong correlation between relevance and usefulness. We see a similar pattern with the Evaluate links.

Operating Systems: The high scores in the Engage usefulness and coverage dimensions does align well with the results obtained in Stage 1 of the evaluation.

Machine Learning: Again, we observe a similar trend in the Machine Learning course as with Operating Systems and Algorithms, except that the coverage scores are slightly lower in this case. One possible reason is that the outline specified by the teacher for some of the lesson plans were really specific and advanced, e.g. "Overfitting, Regularization & Complexity".

Based on the above detailed analysis of 18 lesson plans across three popular subjects in computer science, we observe that the CollectiveTeach 2.0 platform provides sufficient relevant material that covers the four explanatory stages of the 5E model. Across all subjects, we observed a high coverage and utility of the explanatory content retrieved by the system. The Evaluate phase was able to retrieve high quality problems only for a subset of the lesson plans. One of the reasons for this could be the fact that many professors post homeworks/assignments on an internal learning management system, which is not accessible externally on the Web. Another factor was the generic nature of titles used for lessons in specific subjects (for example, the title "Medians and Order Statistics" in Algorithms yielded poor results in the Evaluate phase). In addition, specific lessons in certain subjects also had a limited collection of problem sets that may have been retrievable from the Web. Outside of these specific corner cases, we observe that in a significant majority of lesson plans, the CollectiveTeach 2.0 platform provides high quality results that are usable as an alternative to conventional textbook material.

6.3 Deployment Plans

In order to evaluate the CollectiveTeach 2.0 platform on the ground, we are in the process of deploying and conducting thorough user studies in **Nicaragua** and **India**.

6.3.1 Nicaragua. We shipped a Spanish version of the CollectiveTeach 2.0 for evaluation in Nicaragua. The platform was fine-tuned to generate relevant results in the following subject areas: *English Education* (Spanish Speakers), *Systems Engineering* (Computer Science), and *Fisheries*, and *Nicaraguan History*. Our plan is to test CollectiveTeach 2.0 with professors who teach the above subjects at University of the Autonomous Regions of the Nicaraguan Caribbean Coast (URRACAN).

6.3.2 India. Currently, we are laying the groundwork for a full-scale deployment at an engineering college in the University of Mumbai. The platform will be tested by groups of professors and students in the Computer Science department. The evaluation procedure would test the quality, coverage and appropriateness of the lesson plans generated by the CollectiveTeach 2.0 platform.

7 FUTURE WORK

We discuss here some of the improvements to the CollectiveTeach platform as part of future work. One natural way is to allow the teacher to provide additional terms/keywords to further restrict and improve the quality of the search queries. Our current version of the platform does not personalize the query formulation techniques for individual teachers but this can be easily incorporated in the future. Further, the current version of the platform focused on generating lesson plans only for undergraduate courses, and a natural extension would be including graduate courses as well. However, this will require more sophisticated query formulation techniques, for instance, including research papers and/or projects for advanced graduate courses. We currently do not have any filter for web content that is "old", i.e. university courses that are, say, more than a decade old. However, this can be easily incorporated as part of the automated filtering step and facilitate showing of the most recent educational content to the teachers.

We can also leverage user actions to organically enhance the performance of the platform. For instance, we can have a learning component that keeps track of the search results that were discarded by the teachers and identifies patterns in these results so that they are not queried or are automatically filtered in the future.

Incorporating Student Personalization: The platform presented so far is primarily a tool for teachers to create lesson plans. Though it is possible for the teacher to create different lesson plans for individual (or groups of) students, it is too burdensome and complex a task. Ideally, the platform should be able to generate *personalized lesson plans* for students of different skill levels. CollectiveTeach is designed to generate a sequence of ALUs (as defined in Section 3). Assuming we have access to a measure of "quality" for each ALU, [18] propose a model that enables comparison of different ALU sequences for different students (i.e. having different skill levels) in terms of overall learning. The quality can be determined via feedback (in the form of votes/ratings) either by experts or by users of the platform itself. A promising future direction is to incorporate aspects of that model within the CollectiveTeach framework to automatically compute personalized lesson plans for students of different skill levels.

8 CONCLUSION

To conclude, this paper presented the design, implementation and early deployment experiences of CollectiveTeach, a lesson plan generation platform that enables teachers to collectively organize web-based educational resources within an inquiry-based framework for teaching and learning. Based on a detailed evaluation of 18 lesson plans across 3 undergraduate Computer Science courses, we show that our platform can provide relevant and useful educational resources from the web to help teachers create good quality lesson plans. We also describe the implementation of additional features that make the platform more portable and easy-to-use. In the near future, the platform will be tested on the ground in India and a thorough user study will be conducted to determine the benefits of the CollectiveTeach platform in improving quality of education.

ACKNOWLEDGEMENTS

The authors would like to thank the Center for Technology and Economic Development (CTED) at NYU Abu Dhabi for providing support for the deployment and user study of CollectiveTeach 1.0. In particular, we would like to acknowledge Claudia Amoah Binpong (NYU Ghana), Madonna Kendona-Sowah (NYU Ghana) and Yaw Nyarko (NYU Economics) for their help with the user study. We would also like to thank Kunal Gugale and Abhijai Garg for their effort in building the early version of CollectiveTeach 2.0. Finally, we would like to thank Edwin Reed-Sanchez for fruitful discussions and valuable suggestions for designing the Spanish version of the platform, as well as arranging for a user study in Nicaragua.

REFERENCES

[1] Accessed May 26, 2017. Azim Premji Foundation. http://azimpremjifoundation.org. (Accessed May 26, 2017).
[2] Accessed May 26, 2017. Gooru. http://www.gooru.org. (Accessed May 26, 2017).
[3] Accessed May 26, 2017. Khan Academy. https://www.khanacademy.org. (Accessed May 26, 2017).
[4] Accessed May 26, 2017. Web-based Inquiry Science Environment (WISE). https://wise.berkeley.edu. (Accessed May 26, 2017).
[5] Rakesh Agrawal, Sreenivas Gollapudi, Anitha Kannan, and Krishnaram Kenthapadi. 2012. Data mining for improving textbooks. *ACM SIGKDD Explorations Newsletter* 13, 2 (2012), 7–19.
[6] Rakesh Agrawal, Sreenivas Gollapudi, Krishnaram Kenthapadi, Nitish Srivastava, and Raja Velu. 2010. Enriching textbooks through data mining. In *Proceedings of the First ACM Symposium on Computing for Development.* ACM, 19.
[7] Shaaron Ainsworth and Shirley Grimshaw. 2004. Evaluating the REDEEM authoring tool: can teachers create effective learning environments? *International Journal of Artificial Intelligence in Education* 14, 3, 4 (2004), 279–312.
[8] Ernesto Arias, Hal Eden, Gerhard Fischer, Andrew Gorman, and Eric Scharff. 2000. Transcending the individual human mind-creating shared understanding through collaborative design. *ACM Transactions on Computer-Human Interaction (TOCHI)* 7, 1 (2000), 84–113.
[9] Michael Barbour, Regina Brown, Lisa Hasler Waters, Rebecca Hoey, Jeffrey L Hunt, Kathryn Kennedy, Chantal Ounsworth, Allison Powell, and Trina Trimm. 2011. Online and Blended Learning: A Survey of Policy and Practice from K-12 Schools around the World. *International Association for K-12 Online Learning* (2011).
[10] Rodger W Bybee, Joseph A Taylor, April Gardner, Pamela Van Scotter, J Carlson Powell, Anne Westbrook, and Nancy Landes. 2006. The BSCS 5E instructional model: Origins and effectiveness. *Colorado Springs, Co: BSCS* 5 (2006), 88–98.
[11] Sunandan Chakraborty, Tamali Bhattacharya, Plaban K Bhowmick, Anupam Basu, and Sudeshna Sarkar. 2007. Shikshak: An intelligent tutoring system authoring tool for rural education. In *Information and Communication Technologies and Development, 2007. ICTD 2007. International Conference on.* IEEE, 1–10.
[12] Sunandan Chakraborty, Zohaib Jabbar, and Lakshminarayanan Subramanian. 2015. Summarization Search: A New Search Abstraction for Mobile Devices. In *Proceedings of the 2015 Annual Symposium on Computing for Development.* ACM, 69–70.
[13] Thomas H.. Cormen, Charles Eric Leiserson, Ronald L Rivest, and Clifford Stein. 2001. *Introduction to algorithms.* Vol. 6. MIT press Cambridge.
[14] Michael Crossley and Myra Murby. 1994. Textbook provision and the quality of the school curriculum in developing countries: Issues and policy options. *Comparative education* 30, 2 (1994), 99–114.
[15] Ton De Jong. 2006. Technological advances in inquiry learning. *Science* (2006).
[16] Daniel C Edelson, Douglas N Gordin, and Roy D Pea. 1999. Addressing the challenges of inquiry-based learning through technology and curriculum design. *Journal of the learning sciences* 8, 3-4 (1999), 391–450.
[17] Ýsmet Ergýn, Uygar Kanli, and Yasin Ünsal. 2008. An Example for the Effect of 5E Model on the Academic Success and Attitude Levels of Students':" Inclined Projectile Motion". *Journal of Turkish Science Education* 5, 3 (2008), 47.
[18] Venkataraman et. al. 2017. Traditional, Personalized or Crowdsourced? A Model to Compare Instructional Paradigms. (2017).
[19] John Gillies and Jessica Jester Quijada. 2008. Opportunity to Learn: A High Impact Strategy for Improving Educational Outcomes in Developing Countries. Working Paper. *Academy for Educational Development* (2008).
[20] Paul Glewwe, Michael Kremer, and Sylvie Moulin. 2009. Many children left behind? Textbooks and test scores in Kenya. *American Economic Journal: Applied Economics* 1, 1 (2009), 112–135.
[21] Ricki Goldman, Roy Pea, Brigid Barron, and Sharon J Derry. 2014. *Video research in the learning sciences.* Routledge.
[22] Shalni Gulati. 2008. Technology-enhanced learning in developing nations: A review. *The International Review of Research in Open and Distributed Learning* 9, 1 (2008).
[23] Johann Friedrich Herbart and Charles De Garmo. 1913. *Outlines of educational doctrine.* Macmillan Company.
[24] Ergin I., Uygar K., and Unsal Y. 2008. An example for the effect of 5e model on the academic success and attitude levels of students. *Journal of Turkish Science Education* 5, 3 (2008), 47–59.
[25] Chinwe H Ikpeze and Fenice B Boyd. 2007. Web-Based Inquiry Learning: Facilitating Thoughtful Literacy With WebQuests. *The Reading Teacher* 60, 7 (2007), 644–654.

[26] Chia-Jung Lee, Yi-Chun Lin, Ruey-Cheng Chen, and Pu-Jen Cheng. 2009. Selecting effective terms for query formulation. In *Asia Information Retrieval Symposium*. Springer, 168–180.
[27] David G Levasseur and J Kanan Sawyer. 2006. Pedagogy meets PowerPoint: A research review of the effects of computer-generated slides in the classroom. *The Review of Communication* 6, 1-2 (2006), 101–123.
[28] Marcia C Linn and Sherry Hsi. 2000. *Computers, teachers, peers: Science learning partners*. Routledge.
[29] Susan J Mayer. 2008. Dewey's dynamic integration of Vygotsky and Piaget. *Education and Culture* 24, 2 (2008), 6–24.
[30] Sugata Mitra, Ritu Dangwal, Shiffon Chatterjee, Swati Jha, Ravinder S Bisht, and Preeti Kapur. 2005. Acquisition of computing literacy on shared public computers: Children and the" hole in the wall". *Australasian Journal of Educational Technology* 21, 3 (2005).
[31] James D Slotta. 2004. The web-based inquiry science environment (WISE): Scaffolding knowledge integration in the science classroom. *Internet environments for science education* (2004), 203–232.
[32] James D Slotta and Marcia C Linn. 2000. The knowledge integration environment: Helping students use the Internet effectively. *Innovations in science and mathematics education: Advanced designs for technologies of learning* (2000), 193–226.
[33] Andrew S Tanenbaum. 2009. *Modern operating systems*. Pearson Education.
[34] Randolph Wang, Urvashi Sahni, Sumeet Sobti, Nitin Garg, Jaswinder Pal Singh, Matthew Kam, Arvind Krishnamurthy, and Thomas Anderson. 2005. The digital studyhall. *Computer Science Department, Princeton University, Tech. Rep. TR-723-05* (2005).