Exam Question Recommender System

(Full Paper)

Hicham HAGE       Esma AÎMEUR

Department of Computer Science and Operational Research
University of Montreal
{hagehich, aimeur}@iro.umontreal.ca

Abstract. Although E-learning has advanced considerably in the last decade, some of its aspects, such as E-testing, are still in the development phase. Authoring tools and test banks for E-tests are becoming an integral and indispensable part of E-learning platforms, and with the implementation of E-learning standards, such as IMS QTI, E-testing material can be easily shared and reused across various platforms. With this extensive E-testing material and knowledge comes a new challenge: searching for and selecting the most adequate information. In this paper we propose using recommendation techniques to help a teacher search for and select questions from a shared and centralized IMS QTI-compliant question bank. Our solution, the Exam Question Recommender System, uses a hybrid, feature-augmentation, recommendation approach. The recommender system uses Content-Based and Knowledge-Based recommendation techniques, resorting to the use of a new heuristic function. The system also engages in collecting both implicit and explicit feedback from the user in order to improve on future recommendations.

Keywords: E-learning, E-testing, Assessment tool, E-learning Standards, IMS QTI, Hybrid Recommendation.

Contact Information:

Hicham Hage

Tel: +1-514-343-6794
Fax: +1-514-343-5834

Université de Montréal,
Département d'informatique et de recherche operationnelle,
Pavillon André Aisenstadt
PO BOX 6128,
Station: Centre-Ville,
Montréal Qc, H3C 3J7
Canada
1 Introduction

E-learning has advanced considerably in the last years. Today, there exist many E-learning platforms, commercial (WebCT [11], Blackboard [10]) or open source (ATutor [19]), which offer many tools and functionalities [16]. Some of these tools are aimed towards teachers and developers, and other tools aimed towards students and learners [5]. Although E-learning has come a long way, some of its aspects are still in their early stages. One such aspect is E-testing. While existing E-learning platforms do offer E-testing Authoring tools, most are only basic E-testing functionalities [16] [18], which are limited to the platform itself. With the emergence of E-learning standards and specifications, such as the IMS QTI [14] (IMS Question and Test Interoperability), E-learning material can be reusable, accessible, interoperable, and durable. With E-learning standards, E-testing material can be transferred from one platform to another. Furthermore, some E-learning platforms are starting to offer the functionality of Test Banks. This feature allows teachers and developers to save their questions and exams in the Test Bank for future access and use. To the best of our knowledge, E-learning platforms Test Banks are limited to the teacher’s private use, where each teacher can only access his personal, private questions and tests. Therefore, in order to share available E-testing knowledge, teachers must do so explicitly by using import/export functionalities offered only by some platforms. Consequently, due to the limitations in knowledge sharing, the size of the Test Banks remains relatively small, thus E-learning platforms only offer basic filters to search for information within the Test Bank. In order to encourage knowledge sharing and reuse, we are currently in the works of implementing a web-based assessment authoring tool called Cadmus. Cadmus offers an IMS QTI-compliant centralized questions- and-exams repository for teachers to store and share E-testing knowledge and resources. For such a repository to be beneficial it must contain extensive information on questions and exams. The bigger and more useful the repository becomes, the more dreadful is the task to search for and retrieve necessary information and material. Although there exist tools to help teachers locate learning material [8] [9], to our knowledge there aren’t personalized tools to help the teacher select exam material from a shared data bank. What we propose is to incorporate into Cadmus an Exam Question Recommender System to help teachers find and select questions for exams. The recommender uses a hybrid, feature-augmentation recommendation approach. The first level is a Content-Based filter, and the second level is a Knowledge-Based filter [2] [3]. In order to recommend questions, the Knowledge-Based filter resorts to a heuristic function. Furthermore, the Exam Question Recommender System gathers implicit and explicit feedback [4] from the user in order to improve future recommendations. The paper is organized as follows: section 2 introduces E-learning, E-testing, and offers an overview of E-learning standards, in particular IMS QTI; section 3 presents current recommendation techniques; section 4 describes the architecture and approach of the Exam Question Recommender System; section 5 highlights the testing procedure and the results; and section 6 concludes the paper and presents the future works.

2 E-learning

E-learning can be defined with the following statement: the delivery and support of educational and training material using computers.

E-learning is an aspect of distant learning, where teaching material is accessed through electronic media (internet, intranet, CD-ROM ...) and where teachers and students can communicate electronically (email, chat rooms ...). E-learning is very convenient and portable. Furthermore, E-learning involves great collaboration and interaction between students and tutors or specialists. Such collaboration is made easier by the online environment. For example, a student in Canada can have access to a specialist in Europe or
Asia through email or can assist in the specialist’s lecture through a web conference. There are four parts in the life cycle of E-learning [17]: Skill Analysis, Material Development, Learning Activity and Evaluation-Assessment.

2.1 E-testing
There exist many E-learning platforms, such as Blackboard, WebCT and ATutor that offer different functionalities [16]. Although Evaluation and Assessment is an important part of the E-learning life cycle, E-testing remains in its early development stages. Most E-learning platforms do offer E-testing Authoring tools, most of which offer only basic testing functionalities, and are limited to the platform itself. For instance, most E-learning platforms offer support for basic question types such as Multiple Choice, True/False and Open-Ended Questions, but do not offer the possibility of adding multimedia content (images, sounds …), to set a time frame for the exam, or even include import functionalities to add questions from external sources [16]. In order to deliver E-learning material, each E-learning platform chooses different delivery media, a different platform/operating system and its own unique authoring tools, and stores the information in its own format. Therefore, in order to reuse E-learning material developed on a specific platform, one must change considerably that material or recreate it using the target platform authoring tools—hence increasing the cost of development of E-learning material. Standards and specifications help simplify the development, use and reuse of E-learning material.

2.2 IMS Question and Test Interoperability
As stated in the ADL (Advanced Distributed Learning) goals [13], standards and specifications ensure that E-learning material is: Reusable (modified easily and usable on different development tools), Accessible (available as needed by learners or course developers), Interoperable (functional across different hardware or software platforms), and Durable (easy to modify and update for new software versions). Currently, there are many organizations developing different standards for E-learning [15], each promoting its own standards. Some of the leading organizations with the most widely accepted standards are: IEEE Learning Technology Standards Committee [12], ADL Initiative (Advanced Distributed Learning) [13], and IMS Project (Instructional Management System) [14]. IMS QTI sets a list of specifications used to exchange assessment information such as questions, tests, and results. QTI allows assessment systems to store their data in their own format, and provides a means to import and export that data in the QTI format between various assessment systems.

With the emergence and use of E-learning standards, learning and testing material can be reused and shared among various E-learning platforms [7]. Knowledge sharing would lead to a quick increase in the available information and material, leading to the need for recommendation systems to help filter the required data.

3 Recommender System
Recommender systems offer the user an automated recommendation from a large information space [6]. There exist many recommendation techniques, differentiated upon the basis of their knowledge sources used to make a recommendation. Several recommendation techniques are identified in [2] including: Collaborative Recommendation (the recommender system accumulates user ratings of items, identifies users with common ratings, and offers recommendations based on inter-user comparison), Content-Based Recommendation (the recommender system uses the features of the items, and the user’s interest in these features to make a recommendation), and Knowledge-Based Recommendation (the recommender system bases the recommendation of items on
inferences about the user’s preferences and needs). Each recommendation technique has its advantages and limitations, thus the use of hybrid systems that combines multiple techniques to produce the recommendation. There exist several techniques of hybridization [1] [2] such as: Switching (the recommender system switches between several techniques, depending on the situation, to produce the recommendation), Cascade (the recommender system uses one technique to generate a recommendation, and a second technique to break any ties), and Feature Augmentation (the recommender system uses one technique to generate an output, which in turn is used as input to a second recommendation technique). Our Exam Question Recommendation System uses a hybrid, feature-augmentation approach, using Content-Based and Knowledge-Based recommendation.

4 Exam Questions Recommendation System Architecture
Cadmus is an E-testing platform that offers teachers an extensive question library. The more comprehensive Cadmus’s question library is, the harder the task to search for and select questions. The first suggestion that comes to mind is to filter questions according to their content and the needs of the teacher. A Content-Based filter will help, but might not be enough. For instance, there might be between 50 and 100 questions in the library that satisfy the content requirement, but not all will be rated the same by different teachers with different preferences: a teacher might prefer “multiple choice” to “true and false”, or might prefer questions with a certain level of difficulty. What we propose is a feature-augmentation, hybrid-recommendation approach, where the first level is a Content-Based filter and the second level a Knowledge-Based filter. The Content-Based filter will reduce the search to questions with content pertinent to the teacher’s needs, and the Knowledge-Based filter will sort these questions with regards to the teacher’s preferences, such that the higher ranking questions are the most likely to be chosen by the teacher. Figure 1 illustrates the architecture of the recommender system. We can distinguish two different types of components: Storage components (Question Base and User Profile) and Process Components (Content-Based Filter, Knowledge-Based Filter and Feedback).

4.1 Question Base
The Question Base stores all the questions created by the teachers. The actual question is stored in an external XML file following the IMS QTI specifications, and the database contains the following information about the question:
• **Ident**: unique question identifier
• **Title**: contains the title of the question
• **Language**: corresponds to the language of the question, i.e. English, French …
• **Topic**: denotes the topic of the question, i.e.: Computer Science, History…
• **Subject**: specifies the subject within the topic, i.e.: Databases, Data Structures …
• **Type**: denotes the type of question, i.e.: multiple choice, true/false …
• **Difficulty**: specifies the difficulty level of the question, according to possible values: Very Easy, Easy, Intermediate, Difficult, and Very Difficult
• **Keywords**: contains keywords relevant to the question’s content
• **Objective**: corresponds to the pedagogical objective of the question: Concept Definition, Concept Application, Concept Generalization, and Concept Mastery
• **Occurrence**: a counter of the number of exams this question appears in
• **Author**: the author of the question
• **Availability**: designates whether the question is available only to the author, to other teachers, or anyone
• **QTIQuestion**: handle to the IMS QTI-compliant XML file where the question and all of relevant information such as answers, comments, and hints are stored
4.2 User Profile

The User Profile stores information and data about the teacher that are used by the Knowledge-Based filter. The user profile contains the following:

- **Login**: unique identifier of the user
- **Type Weight**: selected by the user for the type criteria
- **Occurrence Weight**: specified by the user for the occurrence criteria
- **Difficulty Weight**: chosen by the user for the difficulty criteria
- **Author Weight**: specified by the user for the author criteria
- **Individual Type Weights**: system-calculated weight for each different question type, i.e. weight for True/False, for Multiple Selection …
- **Individual Occurrences Weights**: system-calculated weight for each different question occurrence, i.e. Very Low, Average, High …
- **Individual Difficulties Weights**: system-calculated weight for each different question difficulty, i.e. weight for Easy, for Difficult…
- **Individual Authors Weights**: system-calculated weight for each author

![System Architecture](image)

The teacher-specified Type, Occurrence, Difficulty, and Author weights are set manually by the teacher. These weights represent his criteria preference, i.e. which of the four independent criteria is more important for him. The teacher can select one out of five different values with each assigned a numerical value (Table 1) that is used in the distance function explained in 4.4.1. The system-calculated weights infer the teacher’s preferences of the various values each criteria might have. For example, the Type criteria might have one of three different values: True/False (TF), Multiple Choice (MC) or Multiple Selection (MS), thus the system will calculate three different weights: $w_{TF}$, $w_{MC}$ and $w_{MS}$. The system keeps track of a counter for each individual weight (i.e. a counter for True/False, a counter for Multiple Selection …), and a counter for the total number of questions selected thus far by the teacher. Each time the teacher selects a new question, the counter for the total number of questions is incremented, and the corresponding individual weight is incremented accordingly, i.e. if the question is a True/False, then the True/False counter is incremented, and $w_{TF} = \text{Counter (True/False)} / \text{Total number of questions}$. The value of the individual weights is the percentage of usage, so that if the user selected 100 questions out of which 33 were TF, 59 were MC, and 8 were MS, then $w_{TF} = 0.33$, $w_{MC} = 0.59$, $w_{MS} = 0.08$, and $w_{TF} + w_{MC} + w_{MS} = 1$.

<table>
<thead>
<tr>
<th>Weight</th>
<th>Lowest</th>
<th>Low</th>
<th>Normal</th>
<th>High</th>
<th>Highest</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.25</td>
<td>0.5</td>
<td>1</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>
4.3 Content-Based Filter

When, for the purpose of creating a new exam, the teacher wants to search for questions, he must specify the search criteria for the questions (Figure 2). The search criteria are used by the Content-Based Filter and consist of the following: Language, Topic, Subject, the option of whether or not to include questions that are publicly available to students, Objective, Type, Type Weight (used by the teacher to specify how important this criteria is to him, compared with other criteria), Difficulty, Difficulty Weight, Occurrence, Occurrence Weight, Keywords (only the questions with one or more of the specified keywords are retrieved. If left blank, the question’s keywords are ignored in the search), Author (only the questions of the specified author(s) are retrieved), and Author Weight.

The teacher must first select the language and the topic for the question, and has the option to restrict the search to a specific subject within the selected topic. Since some questions may be available to students, the teacher has the option to include or omit these questions from the search. Furthermore, the teacher may restrict the search to a certain question objective, question type, question occurrence, and question difficulty. Moreover, the teacher can narrow the search to questions from one or more authors, and can refine his search further by specifying one or more keywords that are relevant to the question’s content. Finally, the teacher can specify the weight, or the importance of specific criteria (this weight is used by the Knowledge-Based filter). When the user initiates the search, the recommender system will start by collecting the search criteria and weights. Then the search criteria are constructed into an SQL query that is passed to the database. The result of the query is a collection of candidate questions whose content is relevant to the teacher’s search. The candidate questions and the criteria weights are then used as the input to the Knowledge-Based filter.

4.4 Knowledge-Based Filter

The Knowledge-Based Filter takes as input the candidate questions and the criteria weights. The criteria weight is specified by the teacher, and represents the importance of this specific criteria to the user compared to other criteria. Table 1 presents the possible values of the criteria weight and the respective numerical values. The Knowledge-Based filter
retrieves the teacher’s profile from the User Profile repository, and uses the distance function to calculate the distance between each of the candidate questions and the teacher’s preferences.

### 4.4.1 Distance Function

In order to decide which question the teacher will prefer the most; we need to compare several criteria that are unrelated. For instance, how can someone compare the Type of a question with the number of times it appears in exams (the Occurrence)? Since we cannot correlate the different criteria, we left this decision to the teacher: he must select the criteria weight. This weight must either reinforce or undermine the value of the criteria. The Knowledge-Based recommender uses a heuristic Distance Function (Equation 1) to calculate the distance between a question and the teacher’s preferences.

\[
S = \sum_{i} W_i W_j
\]

**Equation 1: Distance Function**

The distance function is the sum of the products of two weights, W and w, where W is the weight specified by the teacher for the criteria and w is the weight calculated by the recommender system. The multiplication by W will either reinforce or undermine the weight of the criteria. Consider the following example to illustrate the distance function: in the search performed in Figure 2, the teacher set \( W_{\text{Type}} = \text{High} \), \( W_{\text{Difficulty}} = \text{Low} \), \( W_{\text{Occurrence}} = \text{Lowest} \) and \( W_{\text{Author}} = \text{Highest} \) (values illustrated in Table 1). Table 2 illustrates the values of two different questions, and Table 3 illustrates the individual weights retrieved from the teacher’s profile. Table 3 contains only a part of the actual profile, reflecting the data pertinent to the example.

<table>
<thead>
<tr>
<th>Question1 (Q1)</th>
<th>Type: True/False</th>
<th>Difficulty: Easy</th>
<th>Occurrence: High</th>
<th>Author: Brazchri</th>
</tr>
</thead>
<tbody>
<tr>
<td>Question2 (Q2)</td>
<td>Multiple Choice</td>
<td>Easy</td>
<td>Low</td>
<td>Brazchri</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Type</th>
<th>Difficulty</th>
<th>Occurrence</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>True/False</td>
<td>Multiple</td>
<td>Easy</td>
<td>High</td>
</tr>
<tr>
<td>Weight</td>
<td>0.33</td>
<td>0.11</td>
<td>0.5</td>
<td>0.06</td>
</tr>
</tbody>
</table>

Calculating the distance function for both questions will give:

\[
s_1 = (W_{\text{Type}} \times w_{\text{True/False}}) + (W_{\text{Difficulty}} \times w_{\text{Easy}}) + (W_{\text{Occurrence}} \times w_{\text{High}}) + (W_{\text{Author}} \times w_{\text{Brazchri}})
\]

\[
s_2 = (W_{\text{Type}} \times w_{\text{MultipleChoice}}) + (W_{\text{Difficulty}} \times w_{\text{Easy}}) + (W_{\text{Occurrence}} \times w_{\text{Low}}) + (W_{\text{Author}} \times w_{\text{Brazchri}})
\]

\[
s_1 = (2 \times 0.33) + (0.5 \times 0.5) + (0.25 \times 0.06) + (4 \times 0.15) = 1.525
\]

\[
s_2 = (2 \times 0.11) + (0.5 \times 0.5) + (0.25 \times 0.54) + (4 \times 0.15) = 1.25
\]

Although there exists a big difference between the Occurrences’ weights in the favor of Q2, Q1 will rank higher because the teacher deemed the Type criteria as more important than the Occurrence criteria.

### 4.5 Feedback

The Exam Question Recommender System first retrieves candidate questions using the Content-Based filter, then ranks the candidate questions using the Knowledge-Based filter, and finally displays the questions for the teacher to select from. The teacher can then select and add the desired questions to the exam. At this stage the exam creation and its effect on
the questions and teacher’s profile is only simulated; no actual exam is created. The Exam Question Recommender System gathers the feedback from the teacher in two manners: Explicit and Implicit. Explicit feedback is gathered when the teacher manually changes the criteria weights, and his profile is updated with the new selected weight. Implicit feedback is gathered when the teacher selects and adds questions to the exam. Information such as the question type, difficulty, occurrence and author is gathered to update the system-calculated individual weights in the teacher’s profile (as highlighted in 4.2).

5 Testing and Results

The purpose of the Exam Question Recommender System is to simplify the task of searching for and selecting questions for exams. The aim of the testing is to determine the performance of the recommendation in helping the teacher select questions. To test the recommender system, we used a database containing about 200 Java questions. The system has a total of 33 different authors/users. For each recommendation and selection, the system recorded the following: Teacher’s Name, Date, Search Number, Questions Recommended, Questions Selected, and Rank. The date and the search number enable us to track the performance and quality of the recommendation as the user makes more choices and his profile is developing. The rank of the selected questions is an indication of the accuracy of the Knowledge-Based Filter, the higher the rank of the selected questions, the more accurate is the recommendation of the Knowledge-Based filter.

5.1 Results

The preliminary results are very encouraging and we are still undergoing further testing. There were 33 registered users (teachers, teacher’s assistants and graduate students) testing the system for a total of 89 recommendations, and 366 questions selected and added to exams (some questions were selected more than once). On average 40 questions were recommended after each search. Figure 3 illustrates the Ranking Partition of the selected questions. Almost 55% of the selected questions were among the top ten recommended questions. Figure 4 illustrates the rank partitioning of the questions selected among the top 10. We notice that the first ranking question is the most selected, while the top five ranked questions constitute about 75% of the selected questions within the top ten ranked by the recommender system. On an average of 40 questions proposed with each search, almost 55% of the selected questions were within the first ten questions recommended by the Exam Question Recommender System, and almost 75% were within the first 20 recommended questions. Thus far, we can conclude that in 75% of the cases, the teacher did not need to browse farther than 20 questions, thereby making it easier for the teacher to search for the required questions for his exam.

![Figure 3: Ranking Partition](image1.png)  ![Figure 4: Top Ten Ranking Partition](image2.png)
6 Conclusion

Today, many E-learning platforms offer authoring tools for E-testing. These authoring tools create E-testing material that will remain mostly confined to the teacher and the platform itself. We are in the process of creating an alternative solution, Cadmus, which offers an independent, IMS QTI-compliant platform to create and share E-testing material. Compared to other platform’s (WebCT, Blackboard, and ATutor) E-testing authoring tools, Cadmus has the advantage of simplifying knowledge sharing. Teachers can choose which material to share, and with whom (Teachers or Students). In addition, since Cadmus stores the questions and exams following the IMS QTI specifications, E-testing material within Cadmus can be easily shared to other E-learning platforms that offer support to IMS QTI and import/export functionality. Furthermore, to help the teachers in their search for information, we proposed the Exam Question Recommender System, which has been tested on a Question Bank of around 200 questions with 33 different users. Preliminary results have shown that the recommendation of the questions is worthwhile. On an average of 40 questions proposed at each search, almost 55% of the selected questions were within the first ten questions recommended by the Exam Question Recommender System, and almost 75% of the selected questions were within the first 20 recommended questions.

What we propose next is to take the Exam Question Recommender System a step further. We propose to enrich the Teacher’s profile to include more information associating the various search criteria. For example, a certain Teacher might associate True/False questions as being “Easy” questions, or prefer the Multiple Selection questions of one author and the True/False questions of another. In addition, by including in the Teacher’s profile information about his approach, methodology, and exam structure, we can create personalized exam templates and then use the Exam Question Recommender System to fill these templates with questions and help automate the exam creation process.

References
[17] http://www.asia-elearning.net/content/aboutEL/index.html
[18] http://www.marshall.edu/it/cit/webct/compare/comparison.html#develop