PerformanceVis: Visual Analytics of Student Performance Data from an Introductory Chemistry Course

Haozhang Deng\textsuperscript{a}, Xuemeng Wang\textsuperscript{a}, Zhiyi Guo\textsuperscript{a}, Ashley Decker\textsuperscript{b}, Xiaojing Duan\textsuperscript{c}, Chaoli Wang\textsuperscript{c}, G. Alex Ambrose\textsuperscript{c}, Kevin Abbott\textsuperscript{c}

\textsuperscript{a}University of Rochester, Rochester, NY 14627, United States
\textsuperscript{b}Fudan University, Shanghai 200433, China
\textsuperscript{c}University of Notre Dame, Notre Dame, IN 46556, United States

Abstract

We present PerformanceVis, a visual analytics tool for analyzing student admission and course performance data and investigating homework and exam question design. Targeting a university-wide introductory chemistry course with nearly 1,000 student enrollment, we consider the requirements and needs of students, instructors, and administrators in the design of PerformanceVis. We study the correlation between question items from assignments and exams, employ machine learning techniques for student grade prediction, and develop an interface for interactive exploration of student course performance data. PerformanceVis includes four main views (overall exam grade pathway, detailed exam grade pathway, detailed exam item analysis, and overall exam & homework analysis) which are dynamically linked together for user interaction and exploration. We demonstrate the effectiveness of PerformanceVis through case studies along with an ad-hoc expert evaluation. Finally, we conclude this work by pointing out future work in this direction of learning analytics research.

Keywords: Student performance, Item analysis, Grade prediction, Learning analytics, Knowledge discovery

1. Introduction

In recent years, more and more efforts have been dedicated to developing learning analytics dashboards to help students, teachers, and other administrative stakeholders gain insights on students’ learning behaviors and performance patterns. These dashboards utilize information visualization techniques to display a variety of information such as the frequency of login, time on task, clickstream, and tool/resource usage by students in an online learning environment. The initial efforts were focused on highlighting potential students at risk of academic failure [1]. More recently, student-facing dashboards were developed to increase students’ self-awareness, promote positive behavior change, and ultimately enhance their academic achievements [2]. However, despite the increasing popularity, the current learning analytics dashboard development has focused primarily on visualizing students’ learning traces, and has not yet developed to adequately promote inclusive teaching and learning or facilitate the improvement of pedagogies and assessment design.

The goal of this study is to present the development process and evaluation outcomes of PerformanceVis, a tool for analyzing and visualizing students’ performance through the lens of time, assessment items, and demographic and academic background. PerformanceVis allows users to analyze and visualize students’ performance in four main views that are integrated via brushing and linking. At the starting point, the \textit{overall exam grade pathway} (OEGP) view presents the change of the grade distribution of different exams in a Sankey diagram. For a detailed investigation of an observed change, users can click on the pathway and bring up the \textit{detailed exam grade pathway} (DEGP) view to identify the student(s) who exhibited the interesting change, their demographic and academic characteristics in more detail. Utilizing the parallel coordinates, DEGP also enables users to filter students by these characteristics and zoom in on their performance trajectory. To validate the assessment design and provide actionable insights on its improvement, we perform exam analyses, including question difficulty and correlation, and present the results in the detailed exam item analysis (DEIA) and overall exam & homework analysis (OEHA) views. DEIA clusters questions in a selected exam by their correlation to each other. When finding an interesting question, users can click on it and investigate how that question is related to questions in other exams and homework via OEHA. Both DEIA and OEHA enable users to analyze exam questions by their difficulty level or embodied topics.

With these four coordinated views, PerformanceVis facilitates inclusive teaching and improves course design. We define “inclusive teaching” as the engagement of underprepared students who scored lower on SAT/ACT and AP tests, and underserved students who were classified categorically by admissions through first-generation, ethnicity, financial aid, and other socioeconomic status. We focus on course design improvement with the intent to increase instructors’ awareness of the potential biases in the assessment, help them eliminate those biases and design reliable and valid exams to assess students’ true mastery of knowledge [3]. In this study, we employ multiple data analysis and visualization methods in combination with...
2. Related Work

Recent studies have revealed that the intended goals of learning analytics dashboards are largely determined by the targeted audiences [5]. Therefore, previously developed dashboards can be generally divided into three categories based on the primary targeted audience: (1) instructor-facing dashboard, (2) student-facing dashboard, and (3) administrator-facing dashboard.

Instructor-facing dashboard. This type of dashboard typically captures and visualizes students’ frequency of login, click-stream pattern, time spent in an online learning environment, as well as their assessment scores and rankings compared to their peers. The primary goal is to assist teachers in identifying students who demonstrate behaviors that may result in low academic achievement. For example, Ali et al. [6] developed LOCO- Analyst to provide instructors with feedback on student learning activities and performance. Similarly, Podgorelec et al. [7] developed Moodle dashboard to help instructors monitor the key indicators of students’ performance within the Moodle Virtual Learning Environment (VLE). To help instructors gain an overview of how students interact with each other in an online learning environment, Bakheria et al. [8] visualized the evolution of participant relationships within discussion forums in SNAPP. Essa et al. [9] developed Students Success System to help instructors identify at-risk students so they could provide timely and effective support to them. As demonstrated by these examples, instructor-facing dashboards aim to help teachers develop an overview of course activity, reflect on their teaching practice, and identify students at risk of academic failure [10].

Student-facing dashboard. On the other hand, student-facing dashboards were also developed to reveal learning patterns to students themselves. This type of dashboard intends to increase students’ self-awareness and promote their self-reflection, with the ultimate goal of motivating them to change their behavior in a way that leads to academic success. Prez-Ivarez et al. [11] developed NoteMyProgress to help students track how they spend time in a course. Santos et al. [12] developed StepUp! to promote student reflection by allowing them to compare their learning activities to that of their peers in open learning environments. Ecoach, developed at the University of Michigan, helps current students pass difficult courses by acquainting them with the feedback and study habits/strategies of previously successful students [13]. Degree Compass is another similar recommendation system for helping current students enroll in courses where they are more likely to succeed by studying the demographics, academic preparation, final grades, and course registration choices of past students [14]. As Klerkx et al. [15] concluded, visualization plays a more versatile role in the educational field than simply increasing information awareness. It has the potential to help shape the learning process and promote reflection on its progress and impact. Accordingly, student-facing dashboards aim to promote students’ metacognition [16] and optimize their academic performance by providing visual displays of their data.

Administrator-facing dashboard. Although the majority of existing learning analytics dashboards are for instructors and students, a few were developed to assist education administrators in strategic decision-making and practice improvement. For example, Krumm et al. [17] developed an early warning system primarily for academic advisers to visualize students’ academic progress and achievement. Charleer et al. [18] developed LISSA to facilitate a data-informed conversation between advisor and student through an overview of study progress and peer comparison. Loughborough University [19] implemented Co-tutor to provide staff with student attendance and performance information so they can monitor student engagement and build relationships with them. Similarly, Student Explorer [20] was developed at the University of Michigan to update academic advisors on their students’ academic performance every
week and help them identify students who need immediate support.

**Instructor- and student-facing dashboard.** Our literature review also discovers that some learning analytics dashboards can directly benefit both instructors and students. For example, Govaerts et al. [21] developed SAM to provide visualizations of course progress for both instructors and students. CCVis, developed by Goulden et al. [22], enables instructors to easily explore the patterns in student course click behavior and identify the course resources that were clicked most and least. It leverages a higher-order network construction algorithm [23] to extract the critical sequences that lead to different transition probabilities, allowing large-scale features to be studied in a node-link diagram. It also correlates the click behavior pattern to grade distributions in a Sankey diagram, which allows users to quickly observe which grades are or are not likely to occur given a specific behavior pattern. This information can motivate and guide students to change their learning behavior to patterns that more likely correlate to better grades. This type of dashboard is not limited to uncovering individual student’s learning behavior in an isolated context. It can also help identify how students form groups and interact with each other in a social network context. For example, NetworkSeer developed by Wu et al. [24] visualizes where, when, and why students interact with each other in MOOC forums. iForum developed by Fu et al. [25] visualizes the three interleaving aspects of MOOC forums (i.e., posts, users, and threads) at three different scales. This enables quick discovery and deep understanding of temporal patterns in MOOC forums.

**Our difference.** We do not find a sufficient number of dashboards that were developed for all of the three categories of audience: instructors, students, and administrators. To bridge this gap, we design and develop PerformanceVis with consideration of the requirements and needs from these three parties. As a result, PerformanceVis provides instructors an overview of students’ perceived difficulty level and topic correlation of exams and assignments. These insights can help them take more targeted action to improve the course and assessment design. Furthermore, PerformanceVis empowers new students to determine which exams and what topics were the most difficult by viewing prior students’ grades distribution and exam item analysis results. With this information, they can make a more efficient and effective study plan. Additionally, PerformanceVis can assist the administrators in evaluating whether or not the S&E program has had a positive impact on scholars’ learning outcomes and help to identify students who need and can benefit from the S&E program the most.

**Machine learning in educational data mining.** Educational data mining extracts valuable information from raw data to achieve better learning processes in courses. In recent years, many machine learning techniques, such as deep neural network, random forest, decision tree, support vector machine, and k-nearest neighbors, have been widely adopted to predict students’ performance [26, 27, 28, 29]. They show better performance than traditional analysis techniques when dealing with plenty of non-linearly separable real-world data with noise. In our work, we use random forest to choose important non-course performance features and build neural networks for the final letter grade prediction.

## 3. Design Requirements

The design requirements are formed based on multiple sessions of discussion with a campus team of learning scientists, designers, and engineers. The primary goal of developing PerformanceVis is twofold. First, it should allow instructors to monitor and manage student performance. In particular, it would be ideal if PerformanceVis could help instructors identify, as early as possible, students at risk of failing, so that they can assist them for possible performance improvement. Second, it should allow instructors to examine their design of homework and exam questions in order to verify their respective roles and identify the connection between coursework design and students’ performance. In particular, it would be ideal if PerformanceVis could help instructors identify any ill-designed questions or low overall design quality for purposeful coursework adjustment or redesign. Therefore, our interface should fulfill the following design requirements to best meet the overall goal.

**R1. Provide an overview of student exam performance trajectory.** The data set used in this research contains performance data from 949 students across 12 homework assignments and 4 exams and as such, it is not very useful to simply display every detail of every student's performance on each item. Instead, the visualization should provide an overview of the data that allows users to quickly and easily understand the overall performance trajectory of students.

**R2. Display grade distribution of different exams.** To understand the general grade trends and how student performance fluctuates throughout the course, we should display the grade distributions for each exam. This should help instructors realize how typical it is for students to “bounce back” from a poor performance or gradually decline throughout the course. This should also satisfy the second objective of evaluating course design by allowing instructors to visualize the rigor of each exam based on their associated grade distributions. For instance, users may find out that many students, even top performers, performed significantly worse on Exam 3. They may ponder whether Exam 3 was a very good discriminator of the top and bottom performers or if it was perhaps too difficult.

**R3. Examine course performance of individual students.** To answer the questions a user may have about a specific student, there must be an option to view the data with greater granularity than just a broad overview. From the overview, users should be able to dive deeper and look at individual students’ data. We should make each student’s data distinguishable from another student so that users may trace specific students throughout the course. The visualization should be able to help instructors extract specific students and look at their characteristics and performance from a detailed perspective.

**R4. Compare course performance of student groups.** As mentioned previously, viewing each pathway for every one of the 949 students would be rather cluttered and users would not be able to derive actionable insights. Thus we should be able to filter the data based on different student groups. We should
display data based on different demographic features, academic features, or grade distributions. In this way, users can compare students who fit similar criteria or contrast different groups of students. This will help instructors identify the characteristics demonstrated by students who got a C or below in the course.

R5. Visualize topic relationships among questions. To address the second objective of evaluating course design, our visualization should represent our item analysis [30] (Pearson correlation, difficulty index, and topic for each question item in the course). We should display the correlations for the question items, especially exam items so that instructors can validate that students are performing similarly on items that were intended to be related (either in topic and/or difficulty). Our visualization of the item analysis should make it more apparent to users how the different exam questions relate to each other and how each homework question relates to the exams. In this way, we will make it possible for users to form a comprehensive course evaluation and gain insights on the improvement of the course design.

R6. Predict student grades. It is of great significance for instructors to monitor and even predict students’ course performance based on their past performance. Doing so will improve the course arrangement and provide extra assistance to students potentially at risk in their study. Obviously, we need to make a trade-off between prediction accuracy and prediction time. Using more grades over time (e.g., after Exam 2 instead of Exam 1) would improve prediction accuracy, but the time remaining in the course limits instructors in making a big difference for those students at risk. Besides, although high accuracy is welcomed, our goal is to predict students’ final letter grades instead of numerical scores, which makes it a classification problem. Furthermore, extra information (i.e., non-course performance features) could likely improve prediction accuracy.

4. Data Analysis

In this section, we briefly introduce the main techniques used to analyze the student performance data gathered from the general chemistry course.

4.1. Pearson Correlation and Linear Regression

A Pearson correlation is simply a number between -1 and 1 that determines the extent to which two different variables (in our case, question items) are linearly related. The closer the value to 1 (-1), the more positively (negatively) correlated the two items. If the value is close to 0, then there is little linear correlation between the two items. We use Pearson correlation to compare students’ performance on different exam and homework questions.

To determine the significance of our correlations, we perform linear regression tests on the correlations. Any p-value below 0.05 indicates the correlation is statistically significant. From our regression analysis, we find correlations above 0.1 are definitely significant and correlations as low as 0.06 could usually still be considered statistically significant. Such low significant correlations, we believe, are a result of the large size of the data. With over 900 records per question item, having 10 or 20 percent of the students perform exactly the same on both items is actually quite significant. The regression analysis not only helps to determine that our data is significant but also helps to determine cutoff correlation values for the information we would display in the visualization.

4.2. Difficulty Index

To aid in our analysis of each exam item’s design, we also calculate the difficulty index for each exam item. A difficulty index essentially describes the percentage of students who got a question item correct [31]. This means the higher the index, the easier the question (or at least the better students performed on that item). The difficulty index helps the instructor determine whether a question is as difficult as it was intended or if it indicates some cultural and language bias that could be eliminated. An accepted range for the difficulty index is typically 30-70%. Any question with a difficulty index below 30% may be considered too difficult and any question with a difficulty index above 70% may be considered too easy. Instructors may want to evaluate the items that fall outside of this acceptable range. With the difficulty index, we can also determine if students were mastering topics if they were able to get the most difficult items under that topic correct.

4.3. Prediction Model

We explore three models for student final grade prediction: a regression model based on a convolutional or recurrent network with course performance features only to determine the optimal timing point for grade prediction, a classification model using a neural network with course performance features only to predict the final letter grade, and a modified classification model which includes the three most important student background characteristics for predicting the final letter grade.

First, we build a regression model using a convolutional neural network (CNN) or recurrent neural network (RNN) to predict students’ final calculated grades. The goal here is to find out the curve of the loss function with the course time passing by, seeking for an ideal timing point to implement the prediction. We consider the following course performance features (scores on 1 extra credit homework assignment: HW 0; 11 homework assignments: HW 1 to HW 11; and 4 exams: Exam 1 to Exam 3 and final exam) and create four timing points (TPs) as follows: TP1 right after Exam 1 (HW 0, HW 1, HW 2, Exam 1), TP 2 right after Exam 2 (HW 3, HW 4, HW 5, HW 6, Exam 2), TP 3 right after Exam 3 (HW 7, HW 8, HW 9, HW 10, Exam 3), and TP 4 right after the final exam (HW 11, final exam). The CNN consists of a convolutional layer followed by a linear layer. The RNN consists of two RNN-cell layers followed by a linear layer. For network training, we set the number of epochs to 1000 and the learning rate to 0.01. We apply the Adam optimizer [32] to update the parameters and use the mean squared error as the loss function. As for splitting the data for training and testing, we use 80% of the data for training and the remaining 20% of the data for testing. Considering that the data set is imbalanced in terms of the final grade distribution, we do
not apply any preprocessing to shuffle the data or to ensure the same distribution between the training and testing data. The results with either CNN or RNN show that the relatively optimal timing point for prediction is TP 1, that is, after Exam 1.

Then, based on four obtained performance features (HW0, HW1, HW2, Exam 1) at the chosen timing point (TP 1), we build a classification model using a fully-connected neural network to predict students’ final letter grades. We consider four coarse categories (A, B, C, and C below) and nine fine categories (A, A-, B+, B, B-, C+, C, C-, and D and below). The neural network consists of four layers, with each one combining a linear layer and a sigmoid activation layer, followed by an extra linear network. For network training, we set the number of epochs to 2000 and keep the remaining parameter values the same as the regression model. We use the cross-entropy loss as the loss function. The splitting of the data for training and testing follows the same way as we do for the regression model.

Finally, we modify this classification model to include three students’ background features such as SAT/ACT score, academic readiness rating, and AP score. We want to measure the significance level of those features in predicting students’ final letter grades and evaluate whether or not including them can improve the prediction accuracy. In Section 6, we compare the results of these two classification models.

5. Visual Interface and Interaction

Our PerformanceVis is a web-based tool for examining student performance, course topic, homework, and exam design of a general chemistry course. Our development utilizes D3.js in order to provide dynamic and interactive visualizations through a web browser. As shown in Figure 1, PerformanceVis mainly includes four coordinated views: overall exam grade pathway (OEGP), detailed exam grade pathway (DEGP), detailed exam item analysis (DEIA), and overall exam & homework analysis (OEHA). In addition, there is a separate view named final grade prediction (FGP) which only shows final grade prediction results to meet the design requirement R6 and is not linked with any other view. OEGP provides an overview of students’ grade distribution over the semester using the Sankey diagram.

DEGP uses a parallel coordinates plot to enable a detailed examination of the student grade data. DEIA depicts the relationship between each question within the same exam by drawing their correlation, which was calculated using the Pearson correlation. Finally, OEHA displays the correlations between exam questions and homework questions along with their topics and difficulty indices in a radial tree structure. These four views are dynamically linked through brushing and linking. In the following, we discuss each of the four main views and its associated interactions in detail.

5.1. Overall Exam Grade Pathway (OEGP)

OEGP meets the design requirements R1 and R2. The goal of OEGP is to visualize the overall trends in students’ grades
throughout the semester. We achieve this goal by using a Sankey diagram to draw an overview of all 949 students’ grade pathways. As shown in Figure 2, the nodes represent five different letter grades, A to F, for four different exams. A flow between two nodes includes the students who received the corresponding grades for both exams. The width of the flow is proportional to the number of students who fall into it. Users are able to find out more detailed information about each flow, including the exact number and proportion of students falling into the flow, by hovering the mouse over the flow.

By looking at the OEGP, users can easily obtain an overview of all students’ performance, and gain insight of general grade trends throughout the course. Furthermore, users could identify specific groups of students who declined or improved over time with the listing of exam grade distributions among all students. Moreover, users could validate the rigor and design of the exams by comparing student grade distributions for different exams.

All flows are clickable. Clicking on a flow will direct users from the student group in OEGP to the same student group in DEGP for detailed investigation.

5.2. Detailed Exam Grade Pathway (DEGP)

DEGP satisfies the design requirements R3 and R4. As shown in Figure 1(c), in DEGP, we utilize a parallel coordinates plot to display each student as a single polyline on a four-dimensional system, where each axis represents an exam of the course, with a horizontal bar chart showing the relative frequency of 19 selected student background characteristics. DEGP serves as a detailed view of the same 949 students shown in OEGP, allowing users to track an individual student’s performance throughout the course. From OEGP to DEGP, DEGP can help users explore the grade composition of a particular flow and the demographic and academic characteristics of the students who demonstrated that grade flow.

Users can trace an individual student’s grade trajectory and his/her background characteristics by mousing over each path. The corresponding student’s characteristics are highlighted in bold under the filter attributes. Users can also compare different groups of students’ performance through filtering the paths displayed on DEGP by their background characteristics or their grades received on each exam. Using the filter checkboxes on the right side of DEGP, users can combine different characteristics and only show the students who satisfy the selected conditions. The bars to the right of each characteristic represent the relative frequency distribution over the data currently showing on DEGP. Different characteristics are distinguished by different colors and users can choose to color the pathways on DEGP by a given characteristic, such as gender. The length of the orange bars to the left of each characteristic represent the predictive importance of each characteristic to a student’s final course grade, measured by the random forest model for predicting whether the final course grade of a student is C or below. It is also possible for users to compare and investigate different student groups by the grades they received. The users can compare up to three student groups in our design by using the grade drop-down boxes below the detailed pathway. For example, we can compare all the pathways for students who received A on Exam 1 to all pathways for students who received B on Exam 1. In this case, the color separates out the student groups who fall into the grade comparisons, not by selected demographic characteristics.

DEGP can help the user, presumably the instructor of the course, examine the most prevalent characteristics among the students who successfully jumped back from a low grade and among students who performed gradually worse. More importantly, DEGP can help to reveal the most common characteristics of students who achieved a C or below course grade in the regular and S&E Scholars cohort.

5.3. Detailed Exam Item Analysis (DEIA)

DEIA fulfills R5 of the design requirements. In DEIA, we use a force-directed graph to show exam questions’ self-correlation, i.e., the correlation between any two questions within the same exam. As shown in Figure 1(a), each node in the graph represents an exam question. The width of the links represents the correlation strength: the thicker the line, the stronger the correlation between the two nodes. The node color indicates either its difficulty level or its topic. DEIA zooms into a specific exam and assesses the questions within that exam in detail.

Using the first drop-down list on the left, users can switch between different exams from Exam 1 to the final exam. In each view, nodes with stronger correlation are clustered together. These clustered nodes will be comprised of similar or related topics. This view can help instructors validate the exam designs by identifying questions that are not supposed to belong to the cluster. Moreover, by hovering the mouse over each node, users are able to find the specific topic and difficulty index value for that particular question, which could also help instructors adjust question type and difficulty level.

5.4. Overall Exam & Homework Analysis (OEHA)

OEHA fulfills R5 of the design requirements as well. By drawing a collapsible radial tree, we can visualize the correlations between exam questions and homework questions all at once. As shown in Figure 1(b), the coarsest level includes the questions of a certain exam currently being displayed in DEIA. For instance, if the final exam is being displayed in DEIA, the coarsest level will consist of the final exam questions, where each node represents a single question of the final exam. The following level displays questions relating to the final exam question node for each of the other three exams. The finest level shows homework questions, revealing which homework questions relate to the exam questions. If Exam 1, 2, or 3 is displayed in DEIA, the coarsest level will be those exam questions and there will only be one level beneath, which shows the homework questions. The preview only displays the very first level of each tree. Furthermore, the size of a node is proportional to the number of children it has. Also, the nodes in OEHA are colored in the same fashion as those in DEIA, showing either the difficulty index level or the topic.

Through brushing and linking, when a node in DEIA is clicked, the corresponding node in OEHA is expanded to show
its position in an overall picture. Users can also manually click the nodes they want to dive into directly from OEHA. By examining the correlation between exam questions and homework questions, or other exam questions, the instructor is able to identify any unexpected strong correlations between two questions, and therefore validate whether certain questions which are not explained well enough in class, are poorly written, or are unrelated to the main material covered.

Figure 2: The OECP view when selecting the flow from C on Exam 3 to A on the final exam.

Figure 3: The grade pathways of students who received C on Exam 3 and A on the final exam.

6. Results and Discussion

Our PerformanceVis is released online at: http://sites.nd.edu/chaoli-wang/demos/. To avoid any compatibility issues (known problems include mousing over OECP showing multiple student records highlighted in black simultaneously), we recommend users to use the Mozilla Firefox browser. In the following, we first present item analysis results. Then, we report four case studies and highlight the insights gleaned. The four studies jointly cover the first five design requirements. After that, we present the final grade prediction results. Finally, we report the evaluation given by a group of experts including learning scientists, designers, and engineers.

6.1. Item Analysis Results

We choose cutoffs for displaying and grouping correlations and difficulty indices based on our linear regression analysis and the data distribution. To avoid visual clutter in the visualization, we do not indicate every single question’s relationship with one another. We also group the difficulty of question items into three categories: easy, medium, and difficult.

For DEIA, we use a correlation cutoff of 0.23 or greater. Each question node will show a connection to another node if their correlation is 0.23 or greater. If a question does not have a correlation of 0.23 or greater with any other questions, we display the connection to the question it has the greatest correlation with.

For OEHA, we use a correlation cutoff of 0.15 or greater to display the first-level correlation between final exam questions and other exam questions. If a final exam question does not have a correlation of 0.15 or greater with any other exam questions, we display the most highly-correlated exam question over 0.12. For the second-level correlation between exam questions and homework questions, we use a correlation cutoff of 0.11 or higher. We only display the top three most highly-correlated homework questions. Any questions with correlations below the aforementioned cutoffs are not displayed to avoid clutter.

We first determine the difficulty index for each question on each exam and then determine cutoffs for the three categories.
of easy, medium, and difficult. The cutoffs for each category vary slightly depending upon the difficulty of each exam. For Exam 1, any question with a difficulty index of 95.5 or above is considered “easy,” any question with a difficulty index less than 95.5 but greater than or equal to 88.5 is considered “medium,” and any question with a difficulty index below 88.5 is considered “difficult.” For Exam 2, any question with a difficulty index between 91 and 81 is considered medium while anything above that range is “easy” and anything below that range is difficult. For Exam 3, any question with a difficulty index between 84 and 64 is considered “medium” while anything above that range is “easy” and anything below that range is difficult. For the final exam, any question with a difficulty index between 87.5 and 77.5 is considered “medium” while anything above that range is “easy” and anything below that range is “difficult.” For the homework questions, no fixed cutoff is set because a large majority of the homework questions have a difficulty index above 0.90. Instead, we evenly divide the homework questions for each homework set into the easy, medium, and difficult groupings, indicating the relative difficulty of questions within a homework set. The questions that fall into the group with the highest, middle, and lowest difficulty indices are considered “easy,” “medium,” and “difficult,” respectively.

6.2. Case Studies

Case study 1: Examining detailed pathway from overall grade distribution. In this very first case study, users need to gain an overview of students’ performance and have a closer look at individual students grade pathways. Then, they would like to compare different groups of students who received different grades on different exams, in order to gain more knowledge of students’ performance on the exams. This case study covers design requirements R1, R2, R3, and R4.
Case study 3: Validating exam questions. This case study shows how users can potentially assess the course design, especially the exam and homework design, by using both DEIA and OEHA together. Design requirement R4 is covered here.

By selecting “Final” from the first drop-down list in DEIA, as shown in Figure 7, it is not difficult for users to identify outliers in this clustered view. By hovering the mouse over the node, users can investigate the detailed information of one of the outliers, final exam question 18.2. This question has a comparatively low correlation, as shown by the thickness of the link, connecting with only one final exam question. From this observation, users wonder whether this final exam question also has low correlation with questions in the other three exams. Users can verify their interpretation by clicking on the question 18.2 node. The corresponding node in OEHA expands and shows that there is only one exam question related to it, as shown in Figure 8, and the only related question is actually about a different topic as indicated by the color difference. Moreover, knowing that the difficulty index value of question 18.2 is 1.00, which means every student got this question correct, users can explain why question 18.2 is an outlier: it is too simple. As a result, users realize that final exam question 18.2 should be
<table>
<thead>
<tr>
<th>View</th>
<th>Visualization</th>
<th>Requirements</th>
<th>Users &amp; Practical Application Questions</th>
</tr>
</thead>
<tbody>
<tr>
<td>OEGP</td>
<td>Sankey diagram</td>
<td>R1, R2</td>
<td>Student/Instructor: Are exam grades being distributed fairly? Which exams are the hardest/easiest? What are the common performance pathways of this course? Can a student recover after receiving a particular score on a specific exam?</td>
</tr>
<tr>
<td>DEGP</td>
<td>Parallel coordinates</td>
<td>R3, R4</td>
<td>Dean/Instructor: How can we reverse engineer and analyze performance of any individual or group of thriving and non thriving students? Are there any course achievement gaps in particular groups of students? Did a particular intervention/treatment have an impact on a special population of students?</td>
</tr>
<tr>
<td>DEIA</td>
<td>Force-directed graph</td>
<td>R5</td>
<td>Instructor: Are there any (or too many) questions to be revised that are too easy, too difficult, or unrelated? Are there enough questions for each topic? Does the question format (multiple choice versus short answer) make a difference?</td>
</tr>
<tr>
<td>OEHA</td>
<td>Radial tree</td>
<td>R5</td>
<td>Instructor: Do the homework questions support and align to exam questions that support later cumulative exams? Do I have enough (too much, or not enough) homework practice problems that align to exam problems?</td>
</tr>
<tr>
<td>FGP</td>
<td>Bar and line charts</td>
<td>R6</td>
<td>Learning scientist: How do we evaluate which predictive model is most accurate at projected grades at certain time points?</td>
</tr>
</tbody>
</table>

Table 1: The summary of the five views of PerformanceVis: their visualizations, requirements, users, and practical application questions.

revised in order to better evaluate whether or not students truly master the material. Through the combined use of DEIA and OEHA, users are able to validate exam questions by looking at their topics, difficulty indices, and relationships with other exam questions.

Case study 4: Analyzing relationship between student performance and coursework design. By now, users already have a thorough understanding of the functions of each of the four views. In this case study, users want to draw the connection between coursework design and students’ performance. This case study covers design requirements R1, R2, and R5.

Users begin with the initial view of OEGP (refer to Figure 2). Looking closely at the grade distribution of all four exams, users note that Exam 3 was the toughest one of the four exams because a significant portion of students received a letter grade of C, D, and F on Exam 3. Knowing this, users immediately want to know what causes Exam 3 to be the most difficult exam. Users then move to DEIA and select “Exam3” from the first drop-down list. DEIA updates itself and displays Exam 3 questions colored with their difficulty levels, as shown in Figure 9 (a). Users can see that questions with the same difficulty level are linked to each other and the result does not deviate from their expectations. But that could not help to explain why Exam 3 was difficult. Thus, users use the second drop-down list to color the nodes in DEIA by topic, resulting in Figure 9 (b). From this view, users see that Exam 3 mainly consists of two topics: Reactions and Thermodynamics. Then users surprisingly discover that almost all of the questions about Reactions are separated far from each other instead of forming a tight cluster, which means that the questions about Reactions may not be well designed. Moreover, even though all the questions about Thermodynamics are connected to each other, they are poorly forming a compact cluster. Therefore, the overall design quality of Exam 3 might be the reason that more students received lower grades on it compared to other exams.

6.3. Final Grade Prediction Results

The results of the final grade prediction are displayed in FGP, which covers the design requirement R6 and is not linked with the other four views. We first report final grade prediction results using the classification model along with four performance features (HW 0, HW 1, HW 2, and Exam 1). With four coarse categories, among 189 students predicted (which is 20% of the data used for testing), 145 students’ grade categories were predicted correctly, 42 students’ grade categories were predicted wrong by one-category difference (e.g., A instead of B), and two students’ grade categories were predicted wrong by two-category difference (e.g., A instead of C). With nine fine categories, 115 students’ grade categories were predicted correctly, and the numbers of students’ grade categories were predicted wrong by one-, two-, three-, four-, and five-category differences are 27, 29, 14, 2, and 2, respectively.

Then, we compare the above final grade prediction results against those obtained using the modified classification model which includes the three most important student background characteristics as well. Our results do not show significant differences in terms of prediction accuracy: for the four coarse categories, the accuracy is 77% (four features) vs. 76% (seven features); for the nine coarse categories, the accuracy is 61% (four features) vs. 59% (seven features). The slight decrease with the addition of three more features is likely because these student background characteristics behave much more like noise for prediction.

6.4. Expert Evaluation

A campus team of learning scientists, designers, and engineers evaluated each visualization by reviewing how well it met the design requirements, outlined how it could be applied,
DEIA. Requirement evaluation: DEIA does an effective job of meeting the design requirement R5 to visualize question item relationships between the exams, homework topics, question type format, and difficulty. User applications: For instructors, DEIA could provide data-driven exam redesign recommendations for future exams and also analyze post-course which topics need to be revisited before a cumulative final. Improvement suggestion: The drop-down menu for DEIA could use labels to better orient users to their purpose. For example, the top drop-down menu could be labeled “Filter by Exam” and the one below it could be labeled “Question Highlighter.” We made the changes as suggested. Additional filter options would be beneficial to users as well. The evaluators propose options like item analysis (i.e., discrimination) or question format (i.e., selective response or constructed response). In addition, it would be helpful to be able to filter by macro-course learning goal to drill down to micro-levels of individual question topics and titles and even possibly to a Bloom’s taxonomy level (i.e., remembering vs. analyzing).

OHEA. Requirement evaluation: OHEA did meet the design requirement R5. For instructors, OHEA could provide data-driven audit and redesign recommendations for overall course syllabus design and topic mapping. User applications: The instructor could use this visualization to provide formative feedback to inform students as to how much time should be spent on a particular topic. A topic identified by OHEA as easy could be allocated less lecture and homework time to allow for topics identified as difficult to be allocated more time. If this visualization were made available to current students it could help assist and customize their study plan. Improvement suggestion: In addition to the suggestions for DEIA, the evaluators propose this visualization should have an open/collapse all levels function with the care that the dynamic tree layout does not get covered by the title of the view.

FGP. Requirement evaluation: FGP does meet the design requirement R6 because it provides an interactive evaluation of which predictive models were most accurate based on historical data. User applications: A learning scientist could use this tool to determine which model and features would best predict current student what-if grades at certain grading points of the course. Improvement suggestion: The new version should include the actual predicted grades for future individual students, not just the accuracy. The charts are missing labels on the x and y axes. We made the changes as suggested.

7. Conclusions and Future Work

In this work, we design, demonstrate, and evaluate PerformanceVis, a visual analytics tool for analyzing student course performance data over time by students’ characteristics and in investigating homework and exam question design. PerformanceVis can help students, instructors, and administrators gain insights into the course. For new students, PerformanceVis helps them preview past student performance, understand how questions from assignments and exams are connected, and get aware of when and where challenging topics would appear in the course.
For instructors, PerformanceVis helps them spot any inconsistency between the intended difficulty level by instructors and the perceived difficulty level by students with respect to question items and adjust their question design accordingly. For administrators, PerformanceVis helps them identify students at risk of failing the course as early as possible and gauge whether the specially designed program meets its goal. Case studies along with the expert evaluation confirm the effectiveness of this learning analytics tool.

Besides updating PerformanceVis based on the remaining suggestions given by the evaluators (Section 6.4), we would like to further improve PerformanceVis in the following two ways. First, it would be ideal to generalize PerformanceVis so that instructors of the same course can quickly reuse the tool to evaluate different offerings simultaneously and instructors of different introductory courses can easily leverage the tool as well. Second, the current version of PerformanceVis analyzes and visualizes pre-collected course performance data. It would be ideal if PerformanceVis can gather the course performance data in real time (i.e., as the semester is in progress) so that important functions such as student grade prediction can be realized in the real deployment of the tool. We would like to address these two issues to make PerformanceVis practically useful.

Acknowledgements

This work was supported in part by the U.S. National Science Foundation through grants IIS-1455886 and DUE-1833129, and the Schlingwein Family Tel Aviv University - Notre Dame Research Collaboration Grant. Haozhong Deng, Xuemeng Wang, Zhiyi Guo, and Ashley Decker conducted this work as an undergraduate research project at the University of Notre Dame during Summer 2019. We also thank the following collaborators: J. Daniel Gezelter, Kelley M. H. Young, Kevin Barry, Mark Gunty, and Catlin Schalk.