UNIVERSITÀ DEGLI STUDI
DI TRENTO

DEPARTMENT OF INFORMATION ENGINEERING AND COMPUTER SCIENCE
**ICT International Doctoral School**

# An Online Peer-Assessment Methodology for Improved Student Engagement and Early Intervention

Michael Mogessie Ashenafi

A dissertation submitted to the
ICT Doctoral School of Università degli Studi di Trento
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Computer Science

**Advisors**
Prof. Marco Ronchetti, Università degli Studi di Trento
Prof. Dr. Ing. Giuseppe Riccardi, Università degli Studi di Trento

**Thesis Examination Committee**
Prof. Gennaro Costagliola, Università degli Studi di Salerno
Prof. Marco Temperini, Università degli Studi di Roma "La Sapienza"
Prof. Vincenzo D'Andrea, Università degli Studi di Trento

November 2017

# Abstract

Student performance is commonly measured using summative assessment methods such as midterms and final exams as well as high-stakes testing. Although not as common, there are other methods of gauging student performance. Formative assessment is a continuous, student-oriented form of assessment, which focuses on helping students improve their performance through continuous engagement and constant measurement of progress.

One assessment practice that has been in use for decades in such a manner is peer-assessment. This form of assessment relies on having students evaluate the works of their peers. The level of education in which peer-assessment is used may vary across practices. The research discussed here was conducted in a higher education setting.

Despite its cross-domain adoption and longevity, peer-assessment has been a practice difficult to utilize in courses with a high number of students. This directly stems from the fact that it has been used in traditional classes, where assessment is usually carried out using pen and paper. In courses with hundreds of students, such manual forms of peer-assessment would require a significant amount of time to complete. They would also contribute much to both student and instructor load.

Automated peer-assessment, on the other hand, has the advantage of reducing, if not eliminating, many of the issues relating to efficiency and effectiveness of the practice. Moreover, its potential to scale up easily makes it a promising platform for conducting large-scale experiments or replicating existing ones.

The goal of this thesis is to examine how the potential of automated peer-assessment may be exploited to improve student engagement and to demonstrate how a well-designed peer-assessment methodology may help teachers identify at-risk students in a timely manner.

A methodology is developed to demonstrate how online peer-assessment may elicit continuous student engagement. Data collected from a web-based implementation of this methodology are then used to construct several models that predict student performance and monitor progress, highlighting the role of peer-assessment as a tool of early intervention.

The construction of open datasets from online peer-assessment data gathered from five undergraduate computer science courses is discussed.

Finally, a promising role of online peer-assessment in measuring levels of student proficiency and test item difficulty is demonstrated by applying a generic Item Response Theory model to the peer-assessment data.

**Keywords**

[peer-assessment, student engagement, early intervention, performance prediction, learning analytics, datasets, higher education]

# Contents

# List of Tables

# List of Figures

# 1

# Introduction

The rapid evolution of Information and Communication Technologies and their widespread adoption by businesses and other institutions towards the end of the previous century have transformed many aspects of our lives. Technological advances have since dictated the creation or ratification of many standards in sectors such as healthcare and finance.

The Internet has been the main driving force behind the evolution of marketing strategies, business models and many other commercial activities. Education, however, is one of a few sectors which have not embraced such technological advances in their entirety. Although ICT has improved efficiency in performing some routine activities that are not directly related to the teaching-learning process, the process itself is seldom transformed by ICT. We are still in an era where the traditional classroom largely remains the gold-standard in the delivery of lectures. More often than not, the teacher is still required to be physically present in a class with a large number of students.

Lack of adoption of educational technology is even more pronounced in higher education. It may be true that face-to-face interaction is an essential part of communication that ICT will never replicate. Nonetheless, there are other problems in traditional education that ICT has the potential to address.

One of these is the issue of assessment, which is more prevalent in higher education settings. This prevalence is due to the typically large number of students that enrol in courses. The usual routine of assigning, collecting, and evaluating assignments in freshman courses with over a hundred students is virtually inexistent, with the exception of many US institutions where such tasks are usually carried out by teaching assistants. In the majority of institutions across the globe, however, students are usually evaluated by their performance in mid-terms and final exams. This is mainly due to the fact that the amount of time and effort required on part of the teacher in disseminating, collecting and grading of assignments increases significantly with the number of students attending the course.

Although the process has been slower than in other disciplines, technological advances are being adopted in the field of education as well. Recent advances in Natural Language Processing (NLP) and Machine Learning (ML) have been used in numerous studies to demonstrate that several levels of text processing can indeed be automated.

Essay grading has also benefited from similar advances. A number of Massive Open Online Courses (MOOCs) such as Coursera already make use of such solutions. Some commercial solutions also use a combination of human- and machine-assigned scores. A prominent example is the essay grading technology used by Educational Testing Service (ETS).

The issue of plagiarism in higher education has also been addressed using NLP and ML techniques. Turnitin, for instance, is a commercial plagiarism checking solution that is currently in use by many online learning platforms.

Other applications of ICT in education relate to transforming or reshaping pedagogical models using multimedia technologies, which are probably the most common form of technology in all levels of education. The Flipped Classroom approach, for instance, makes use of online learning platforms and short video lectures to transform the role of instructors into tutors.

Assessment and evaluation of student performance is one of the main topics in education where alternative solutions to traditional forms of assessment had been proposed and experimented with long before computers were even powerful enough to allow interaction through Graphical User Interfaces. Although the goals may vary, several alternatives to the teacher being the sole assessor of students are now in use. Two of these are self-assessment and peer-assessment. Broadly put, the former focuses on how to improve student learning through the student's reflection on or assessment of their own work. The latter, puts more weight on how students could learn by providing feedback on their peers' work as well as learning by incorporating feedback from their peers.

Several opportunities exist for transforming these alternative forms of assessment. In particular, the work discussed in this thesis focused on semi-automation of peer-assessment and how to take advantage of the opportunities that arise from this automation. These opportunities are related to learning strategies used by students as well as choices of pedagogy for the teacher.

## 1.1 Motivation

Higher education dropout rates are generally high across the world, even in developed nations. The National Center for Education Statistics (NCES) of the US Department of Education put the rate of those first-time, full-time undergraduate students who do not graduate within 6 years from a 4-year degree granting institution at 40% for the fourth quarter of 2008 [48]. While northern and western Europe have a much lower rate, Italy has one of the highest university dropout rates, with only 32% of 2012's Italian youth expected to complete university education in their lifetime [67].

A 2015 report by the Organisation for Economic Co-operation and Development (OECD) identifies several factors that have led to high dropout rates in higher education across European universities [67]. These may be factors at the national level such as education policies, tuition fees or student financial aid. Other factors at the individual level are explained by the OECD report as relating to family or socioeconomic background, gender, ethnic background, cognitive competencies and motivational disposition of the student.

A study involving 6000 students across 18 baccalaureate-granting US institutions found that engagement had positive, statistically significant effects on retention and student success, especially at the first two-years of college [54].

The main motivation behind this thesis was, therefore, exploring ways to address the problem of high dropout rates from the perspective of improved student engagement as a result of application of computer science solutions. The aim was to design solutions that, in particular, focused on monitoring competencies and improved student engagement.

To this end, the work discussed in this thesis involved several batches of students enrolled in first and second year undergraduate-level computer science courses at the University of Trento, over a period of four years.

Peer-assessment is a practice with goals that have very much in common with the motivations of this work. It is for this reason that peer-assessment was chosen as the appropriate pedagogical model for fostering student engagement and performance monitoring.

As a pedagogy, peer-assessment faces its own problems related to both effectiveness and efficiency. The work discussed in this thesis addressed these problems through the introduction of an online peer-assessment system that automated the majority of activities carried out by students and the teacher.

Automation of such tasks had returns that improved efficiency and effectiveness. Use of the online peer-assessment platform by students led to continuous generation of data about student activity. These data were, in turn, utilized to build models that allowed monitoring student progress and early prediction of expected student performance.

Prediction of student performance is not limited to peer-assessment. Indeed, student performance prediction uses several sources of data and their combinations to predict various outcomes. Dropout is one of the most common outcomes that automated prediction systems focus on. Data from intermediate quizzes, midterms, take-home assignments and achievements in earlier years of school have all been used to make predictions.

A number of studies use peer-assessment data to predict dropout in MOOCs. Peer-assessment activities may be designed as small mini-tasks that require relatively small amount of time to complete. This would enable students to carry out several mini-tasks in a continuous manner. This continuous stream of data could then be used to predict more specific outcomes than just dropout, such as expected performance levels of students.

Peer-assessment data from previous batches of students may also be used to construct models that could make early predictions to identify those who may be at risk of dropping out. Such tools of intervention are especially important in courses with a large number of students, where the instructor may have neither the resources nor the time to closely monitor students.

Moreover, automated peer-assessment may be seen as having the auxiliary role of shifting instructor load to students. In the Italian higher education system, for instance, the concept of graduate assistants, who would carry out the professor's voluminous repetitive tasks, is largely nonexistent. Automated peer-assessment, has the additional goal of distributing assessment loads over to students, given that the assessment tasks are well-defined and structured. This, however, has very rarely been the focus of peer-assessment studies, probably due to concerns regarding the validity and reliability of peer-assessment itself, expressed in many studies.

Other motivations to explore additional roles of automated peer-assessment emerged through the course of this study. These were investigation of its potential as a tool of early intervention and how data from automated peer-assessment platforms could be used to model test items and student proficiencies.

Measuring the true effect of a proposed strategy to a problem, especially when the problem has to do with the teaching-learning process, requires observation of the effects of the strategy over a period of time well beyond the span of this research work. It is, nonetheless, hoped that this work will be the first attempt at realizing a novel, technology-supported peer-assessment, which future work could build upon.

## 1.2 Thesis Goals

The main goal of this thesis is to demonstrate the advantages of automating peer-assessment practices and to explore how opportunities that are brought about by such automation can be made full use of.

The two main hypotheses of this thesis are:

- A well-designed online peer-assessment methodology can promote student engagement and
- Data from such methodology can serve as a tool of early intervention by predicting student success and identifying at-risk students in a timely manner.

Automated peer-assessment tasks could promote student engagement and serve as good indicators of student performance. The approaches demonstrated in this study foster student engagement by encouraging continuous participation in online peer-assessment tasks. As a result, a significant correlation may be established between participation in carefully designed online peer-assessment tasks and student performance in summative assessment tasks such as end-of-course exams.

An extended goal of this thesis is to apply principles of Item Response Theory (IRT) in order to model the quality of students' expected responses to questions provided by their peers.

## 1.3 Thesis Contributions

This thesis is interdisciplinary in that it brings together the fields of education and computer science. Motivated by recent successes in applications of computer science in sectors such as healthcare, it aims to address problems in peer-assessment in higher education and create new opportunities by applying solutions from computer science and engineering.

This thesis is novel in that it advances to a further extent than before efforts to apply machine learning and software engineering solutions that revitalize peer-assessment practices, both in technical and pedagogical aspects.

The contributions of this thesis are:

1. An online peer-assessment platform that enhances student engagement throughout a course
2. A novel design of peer-assessment tasks that advances the role of peer-assessment as a tool of early intervention in higher education settings
3. A linear-regression model trained with peer-assessment data for predicting end-of-course student performance
4. Peer-assessment driven linear regression models that can trace and predict student progress within a few weeks of a course's start
5. A novel attempt to model student proficiencies and test item difficulties using questions submitted by students.
6. Publicly available peer-assessment datasets

## 1.4 Structure of the Thesis

This thesis is structured as follows.

- **Chapter 1 - Introduction** - This chapter provides a description of the motives and reasons behind this thesis. It introduces the reader to the main goals of this thesis and establishes the topics on which it focuses. It explains in more detail the contributions of this thesis and how the work presented is structured.
- **Chapter 2 - Literature Review** - As discussed earlier, this thesis has an interdisciplinary nature. It adopts the practice of peer-assessment in higher education as the platform on which the thesis work is built. It explores the applicability of machine learning techniques to peer-assessment data to predict expected student performance. Therefore, this chapter provides a review of recent literature in both peer-assessment in higher education and student performance prediction.
- **Chapter 3 - Peer-Assessment for Promoting Student Engagement** - This chapter is dedicated to the discussion of the web-based peer-assessment platform that has been the foundation of experiments that spanned four years. These experiments were conducted in actual classes, where students actively used the online peer-assessment platform. Results of two rounds of student surveys that confirmed the importance of the tasks in enhancing student engagement are reported here.
- **Chapter 4 - Online Peer-Assessment as a Tool of Early Intervention** - This chapter builds upon the findings reported in the previous two chapters and studies the correlation between participation in online peer-assessment tasks and successful course completion.
- **Chapter 5 - Predicting Student Success from Peer-Assessment Activities** - This chapter explores how online

peer-assessment data can be used to train a linear regression model that could predict final exam scores of students between the range 18-30. It further argues how predicting a range may be beneficial and proceeds to predict grades, with significant improvement in performance.

- **Chapter 6 - Monitoring Student Progress from Peer-Assessment Activities** - This chapter focuses on making modular and continuous predictions of student performance. It discusses weekly predictions of student performance over an eight week period for two courses. It proposes several interpretations of progress and what this series of predictions may look like in the case of two of these interpretations. It demonstrates that online peer-assessment tasks could trace student progress with small degree of error within the first few weeks a course.

- **Chapter 7 - Estimation of Student Proficiency and Test Item Difficulty from Peer-Assessment Data** - This chapter explores whether the quality of students' answers to peer-submitted questions, expressed in terms of peer-assigned marks, could fairly indicate the difficulty level of the questions. Moreover, it studies whether such peer-assigned marks could be used to model the proficiency of students. To this end, the linear regression models discussed in earlier chapters are used in combination with one of the most common response modelling approaches, Item Response Theory (IRT), in order to build an IRT model. A validation framework is used to evaluate the performance of the model.

- **Chapter 8 - Online Peer-Assessment Datasets** - This chapter describes the construction of the datasets used in this study. It proposes experiments that may be conducted using these datasets.

- **Chapter 9 - Discussion and Conclusion** - This chapter summarizes the main objectives of this thesis, the studies conducted in order to achieve them and to what extent they were met. It provides a summary of the work discussed in each of the chapters and how it relates to the main goals of this thesis. Challenges encountered throughout the course of this work and measures taken to overcome them are discussed. A prospect of work yet to be carried out and opportunities for future research constitute the closing remarks of this chapter.

## 1.5 Relevant Publications

- Ashenafi, M. M., Riccardi, G., & Ronchetti, M. (2014, June). A Web-Based Peer Interaction Framework for Improved Assessment and Supervision of Students. In *World Conference on Educational Multimedia, Hypermedia and Telecommunications* (No. 1, pp. 1371-1380).

- Ashenafi, M. M., Riccardi, G., & Ronchetti, M. (2015, October). Predicting students' final exam scores from their course activities. In *Frontiers in Education Conference (FIE), 2015. 32614 2015. IEEE* (pp. 1-9). IEEE.

- Ashenafi, M. M., Ronchetti, M., & Riccardi, G. (2016, July). Predicting Student Progress from Peer-Assessment Data. In *9*sup*th International Conference on Educational Data Mining, International Educational Data Mining Society.*

- Ashenafi, M. M., Ronchetti, M., & Riccardi, G. (2016, October). Exploring the Role of Peer-Assessment as a Tool of Early Intervention. In *ICWL 2016 - 15*sup*th International Conference on Web-based Learning.*

- Ashenafi, M. M., Riccardi, G., & Ronchetti, M. (2016, November). Using Students' Collaboration to Improve Active Participation to University Courses with Large Number of Attendees. In *9*sup*th annual International Conference of Education, Research and Innovation.*

- Ashenafi, M. M. (2017). Peer-assessment in higher education–twenty-first century practices, challenges and the way

forward. *Assessment & Evaluation in Higher Education*, 42(*2*), 226-251.

- Ashenafi, M. M. (2017). A Comparative Analysis of Selected Studies in Student Performance Prediction. *International Journal of Data Mining & Knowledge Management Process*, 7(*4*), 17-32.

# 2

# Literature Review

## 2.1 Introduction

As stated in the previous chapter, the work discussed in this thesis is interdisciplinary in that it brings together the practice of peer-assessment in higher education and student performance prediction using a machine learning approach.

It is, hence, important that recent work in both peer-assessment and student performance prediction are reviewed beforehand. Accordingly, this chapter dedicates one section to the review of each area of research.

## 2.2 Peer-Assessment in Higher Education

Assessment and evaluation of students in higher education settings mostly follow a summative format, where the extent to which students have achieved specific learning goals is commonly measured at specific intervals throughout a course [41, 64, 65]. Typical summative assessment tasks include midterms, final exams and written assignments that are submitted as parts of a coursework. Both criterion-based and norm-referenced forms of summative assessment are in use. The first form establishes a student's performance by determining whether specific and clear public standards are met, whereas the latter evaluates a student's standing relative to the achievements of other students in the same cohort [41, 64].

Doubts about the reliability and effectiveness of summative assessment have been cast by scholars such as Boud and Knight, who highlighted the importance of feedback and argued that goal of assessment should be promoting learning [50, 14].

Formative assessment is a form of assessment that is built on top of such arguments. It is intended to provide feedback and support to students so they could monitor their own progress and identify their strengths and weaknesses. Formative assessment should incorporate detailed feedback and should not contribute towards final marks.

One non-traditional assessment approach, which commonly adopts formative assessment is peer-assessment. In this form of assessment, students or groups of students assess the woks of their peers. A formal definition provided by Topping for peer-assessment is '... an arrangement in which individuals consider the amount, level, value, worth, quality, or success of the products or outcomes of learning of peers of similar status [89].

The advantage of feedback and forms of assessment that heavily rely on it could, however, quickly diminish with growth in class size. The significant, and barely manageable, increase in instructor workload that would be incurred by introducing formative assessment and detailed feedback in university courses could by itself deter the adoption of the method.

The same is true for peer-assessment, at least in higher education settings. Implementing peer-assessment in a course usually requires the extra effort of carrying out repetitive tasks such as the design, distribution and collection of peer-assessment tasks. The manual nature of such tasks thus makes the practice an unrealistic candidate for formative assessment in classes with a large number of students.

Peer-assessment has been used in higher education settings for over half a century. Research has since shown that its manual nature is one of the many factors that determine its efficiency and effectiveness. Perhaps the most influential works that consolidate findings of research conducted before the turn of the century are those carried out by Topping [89] and Falchikov and Goldfinch [27].

Topping identified several variables that determined the effectiveness of peer-assessment, which varied among the 109 studies that he reviewed. Variation in curriculum areas, objectives of peer-assessment projects, whether peer-assessment was conducted in summative or formative settings, the type of work being assessed, degrees of agreement between peer- and teacher-assigned scores were among the factors he identified.

Consequently, Topping concluded that the existence of too many variables across the studies he examined meant that it was difficult to establish whether peer-assessment was either a sound or practical approach in higher education settings.

Two years later, Falchikov and Goldfinch argued that a meta-analytic approach needed to be followed to study whether peer-assessment was a reliable and valid approach. The meta-analytic approach they applied to over fifty studies that compared peer- and teacher-assigned marks identified population characteristics, the work being assessed, the course level, the nature of assessment criteria, and the number of teachers and students involved in peer-assessment tasks as the main variables that affected the quality of the studies. They concluded that, on average, peer marks agreed with teacher marks. They identified six factors that would most likely influence improvements in agreement between peer- and teacher-assigned marks. They concluded that:

- Studies in which well-specified criteria were provided to students and they were asked to provide overall judgments instead of specific judgment per criteria had better peer-teacher mark agreements.

- Educational peer-assessment studies seemed more effective than those conducted in professional settings.

- Better experimental designs led to better agreements.
- There was a weak relationship between increase in the number of peers per single assessment task and decrease in score agreements.
- Medical subject areas tended to have less peer and teacher score agreements.
- Better agreements between peer and teacher scores were achieved when students were involved in the definition of clearly stated scoring criteria.

Recent studies in peer-assessment have focused on a number of themes including peer-feedback, design strategies, student and teacher perceptions, social and psychological factors, student engagement, variables and qualities across studies, design strategies and validity and reliability of the practice.

Validity, which refers to score agreements between peers and teachers, and reliability, which refers to closeness of scores assigned by multiple peers, are still the most studied factors. However, results of the large number of studies conducted in this area have not managed to either support or reject the hypothesis that peers are reliable markers. The influential study by Falchikov and Goldfinch found an average correlation of 0.69 for the fifty-six studies. Nonetheless, this degree of correlation, although high, is not strong enough to warrant definitive substitution of peer marks for those of teachers.

The settings and variables of peer-assessment tasks in the studies considered by Falchikov and Goldfinch varied with experiments. This was especially the case for sample sizes and number of students involved per assessment task. Automated peer-assessment environments could help alleviate problems related to validity and reliability in a number of ways. Before making the case for automated peer-assessment, however, formal definitions of both traditional or manual and automated peer-assessment are offered.

Traditional (Manual) Peer-Assessment practices are those that do not utilize electronic equipment such as Electronic Voting Systems (EVS) or clickers such as those used in peer-instruction [26, 81] or information technology artifacts such as computer software in order to improve the efficiency and effectiveness of processes. In such practices, the work to be assessed is either hand-written or orally presented. The collection and assignment of the work to be assessed is carried out manually. Students rate and comment on their peers' works by providing either oral or hand-written feedback. The specification and communication of criteria, if any, takes place in the form of traditional classroom discussions or lessons.

Automated Peer-Assessment on the other hand may utilize electronic equipment or information technology artifacts to automate, partially or entirely, the processes involved. Typical automated peer-assessment environments use computer software to facilitate the distribution and delivery of assessment tasks as well as the completion of tasks and communication of assessment results. Advanced computer science solutions such as NLP and ML may also be used to grade essays. These grades may be used to calibrate peer-assigned marks. Some automated peer-assessment tools also provide teachers the option to specify assessment criteria to be used by students when assessing their peers' works. Semi-automated peer-assessment refers to practices in which only certain processes such as distribution and collection of assessment tasks are automated.

Regarding problems of validity and reliability, the case for automation is made through the identification three potential improvements.

Firstly, automation of peer-assessment tasks improves scalability of experiments by automating redundant and time-consuming activities such as distribution and collection of assignments that do not necessarily contribute to the objectives of the practice itself. With such activities that prohibit large-scale experiments out of their way, researchers could then conduct several cycles of new experiments or replicate previous ones with much larger cohorts of students, for roughly the same amount of time it would take to conduct manual versions of those experiments. Therefore, automation offers sustainability to the practice and allows extensive research that could eventually lead to a widely shared consensus regarding the practice's validity and reliability.

Secondly, automation could pave the way to efficient analysis of large amounts of teacher and student activity data collected over several runs of experiments. Data mining techniques could be deployed in automated environments to extract important information such as time spent on completing assignments and other online activity, which might otherwise be unavailable or difficult to obtain.

Thirdly, automated calibration of peer scores according to information automatically extracted from repeated runs of peer-assessment experiments could help adjust peer-assigned scores to improve agreement with teacher scores (Hamer et al. 2005). Automated Essay Scoring (AES) and Calibrated Peer Review (CPR) both take advantage of automation. Massive Open Online Courses (MOOCs) usually apply one of these techniques to assess students' works. CPR relies on rating abilities of students, which are determined through an initial assessment, to calibrate peer-assigned scores. AES, on the other hand, solely uses ML and NLP techniques to rate essays. While both approaches are not necessarily mutually exclusive, most notable MOOC providers such as Courseera and EdX have opted to integrate only one of them into their courses [11]. Automation is the path to applying any, or a combination, of these approaches to peer-assessment tasks.

Apart from validity and reliability, most literature reviews in peer-assessment have focused on factors such as student involvement, identification of variables of peer-assessment, and quality factors. Student involvement is one of the most important factors that determine the effectiveness of peer-assessment [28, 87]. According to Bloxham and West (2004) [13] and Sluijsmans et al. [83], student involvement should go beyond participation in assessment tasks; students should take part in the specification of assessment criteria as well.

As the factors that have been of more importance in peer-assessment studies over the years are unveiled, it also becomes apparent that many of the requirements for setting up the ideal peer-assessment environment become more and more difficult to meet as one moves from a class with a handful of students to large classes, such as those common in freshman courses. The need for efficiency becomes stronger than ever, so does the case for automation.

Student involvement has been addressed by several studies, which recommended that students be actively involved in the various stages of peer assessment. Falchikov argued that any assessment task must have students as active participants in order for it to be effective, should allow replication and provide students with clear instructions regarding the processes involved. The importance of student involvement in all stages is also highlighted by Tillema et al. [87], while the importance of involving students in the specification of assessment criteria has also been stressed in other studies [13, 83].

One of the largest manual peer-assessment studies so far was a three-phase study conducted over a two-year period, involving 1654 students and 30 staff from three departments [12]. This study demonstrated how rigorous, long-term peer-assessment could be conducted and followed an action research process to design peer-assessment procedures. Students were involved in the development of clearly stated criteria that were subsequently revised by eliciting their continuous involvement.

While the study had high quality and was exemplary, it lacked attributes that would promote sustained implementation of the proposed approach. The distribution of assignments to peers was manual, and given the high number of students involved, such was the effort needed to implement anonymous peer assessment that some departments subsequently opted to forgo anonymity. Lack of anonymity is perceived by students as a negative factor that deters participation in peer-assessment tasks. It undeniably increases the risk of bias.

Once more, the argument for automating peer-assessment is strengthened by the fact that automation seamlessly introduces anonymity and random task assignment features into the process.

While an overall positive perception of students towards peer-assessment has been reported by some studies [69, 75, 95, 96, 56, 21, 98, 60], a survey of 1740 students and 460 faculty involved in peer-assessment found that most students and faculty perceived summative peer-assessment as ineffective due to concerns about the ability of students to rate their peers [57].

Whether students' criticism of their peers' ability has truth or arises from bias can be well-tested in a peer-assessment environment that maintains anonymity. A possible scenario is where a teacher plays the role of a student and assesses 'peers' in an anonymous experiment where students are not notified of the teacher's involvement. Changes in opinions of students, or otherwise, after the conclusion of the experiment should provide enough information to accept or reject the null hypothesis that students are not unreasonably critical of their peers' ability to assess their work. An automated peer-assessment environment would be a perfect candidate for this experiment.

The number of studies comparing peer and teacher marks has steadily increased since the work by Falchikov and Goldfinch. In this thesis, a meta-analytic review of fifteen studies published since then was conducted. The review found similar results to those of Falchikov and Goldfinch. In addition to the attributes reported by Falchikov and Goldfinch, the review introduced contribution of peer-assessment marks towards final grade and anonymity as two other attributes of the studies. After excluding studies that did not report comparable results, the average correlation between peer and teacher marks for the eight remaining studies was found to be high (r=0.8). For five of the studies, the data reported was used to compute an effect size (d) of 0.27, which implied strong agreement between peer and teacher marks (when comparing peer and teacher marks, smaller effect sizes are sought [27]). Detailed analysis of these studies and an extensive review of 64 peer-assessment studies published since 2000 is provided in [6].

Many peer-assessment studies are conducted in traditional classroom environments as one-off experiments that are rarely replicated in similar settings. Moreover, the opportunities provided by advances in similar practices such as collaborative learning have seldom been taken advantage of [51]. As has been proposed throughout this review, applied computer science could help alleviate much of the problems related to efficiency and effectiveness of the practice. Lack of important collaboration of PA practitioners with researchers in other fields is seen as hindering the sustained development of the practice [51, 85].

ML approaches have been used in other educational settings to predict student performance but application of statistical methods in peer-assessment has so far focused on developing weighting mechanisms to calibrate peer-assigned scores [91, 34, 11, 17].

The work discussed in this thesis introduces two new dimensions. The first is improving student engagement through automated peer-assessment. A relationship was established between lack of participation in online peer-assessment tasks and course incompletion. Chapter four explores how this relationship was established and proposes automated peer-assessment as a tool of early intervention.

The other dimension is using prediction models in peer-assessment as tools of student performance and progress monitoring. This dimension is demonstrated in chapters five and six. Before discussing how prediction models could be applied to peer-assessment data, however, a review of their application in other educational settings and a discussion of their potential to be applied in settings such as peer-assessment is in order.

## 2.3 Student Performance Prediction in Higher Education

In educational settings, performance prediction is carried out with the intention of providing students and teachers information that can be used to measure progress and to identify students at risk of failing well ahead of time so that appropriate measures are taken both on parts of the teacher and the student to avoid such risk. For this reason, timeliness shall be regarded as a necessary albeit hardly sufficient condition for the effectiveness of performance prediction and other prediction models in general.

The majority of studies in performance prediction have been conducted in higher education settings and apply one or more machine learning algorithms to build performance prediction models. For this reason, this review focuses on studies in higher education that applied machine learning techniques to student performance data collected in traditional or online learning environments, or both.

### 2.3.1 Evolution of Data Sources in Student Performance Prediction

Earlier research focused on determining whether standardised test results that were obtained at earlier levels of education could predict student success in later years of higher education [30, 25, 44, 3]. Most recent studies do not incorporate student performance data from lower levels of education. Those that do so incorporate additional parameters obtained from more recent achievements such as freshman and sophomore years [42, 55, 88, 5].

Recent performance prediction studies are refined in the parameters they utilize, the statistical measures they employ, and the outcomes they predict. Demographic data, performance on take-home assignments, projects, and activity on online learning platforms have all been used to build performance prediction models. **Appendix A** provides a list of performance prediction studies with details of the data used, course levels, algorithms, number of students, and evaluation results.

The Internet has made it possible to construct online interaction environments, paving the way for the collection of student data in an unparalleled manner. MOOCs are a perfect example of this, where data from tens of thousands of students is used to train statistical models [10, 74]. Accordingly, recent research in performance prediction has augmented traditional parameters with online student activity data. This phenomenon has helped ease the transition from prediction of a binary outcome such as a pass or a fail to prediction of more fine-grained outcomes such as grades. Examples of studies that utilized online student activity data in predicting performance include [2, 7, 8, 80, 52, 37, 59, 103].

Among 46 studies that explored the application of performance prediction in higher education settings, 70% used at least two student performance data sources to build their prediction models. Current semester performance was the most used and high school data, the least. Figures 2.1 and 2.2 provide detailed information about the data sources. The chart in figure 1, therefore, shows a higher number than the actual 46 studies because it reflects use of multiple data sources by 70% of the studies.

Figure 2.1: Number of Studies per Data Source

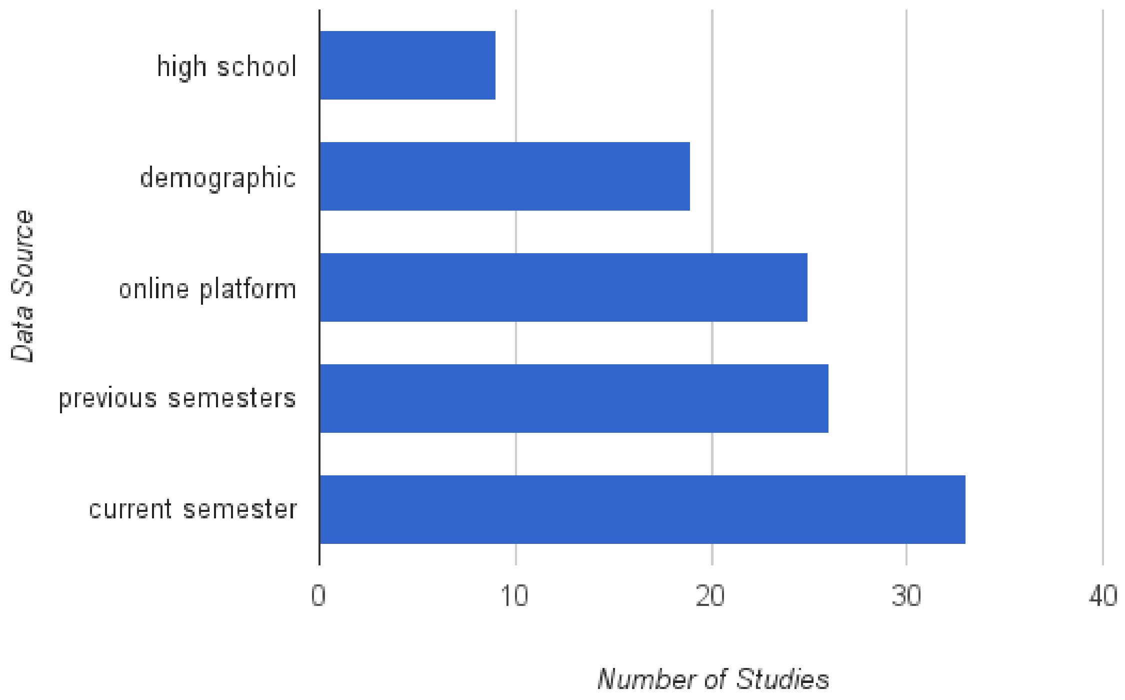


Figure 2.2: Data Source Utilisation per Percentage of Studies

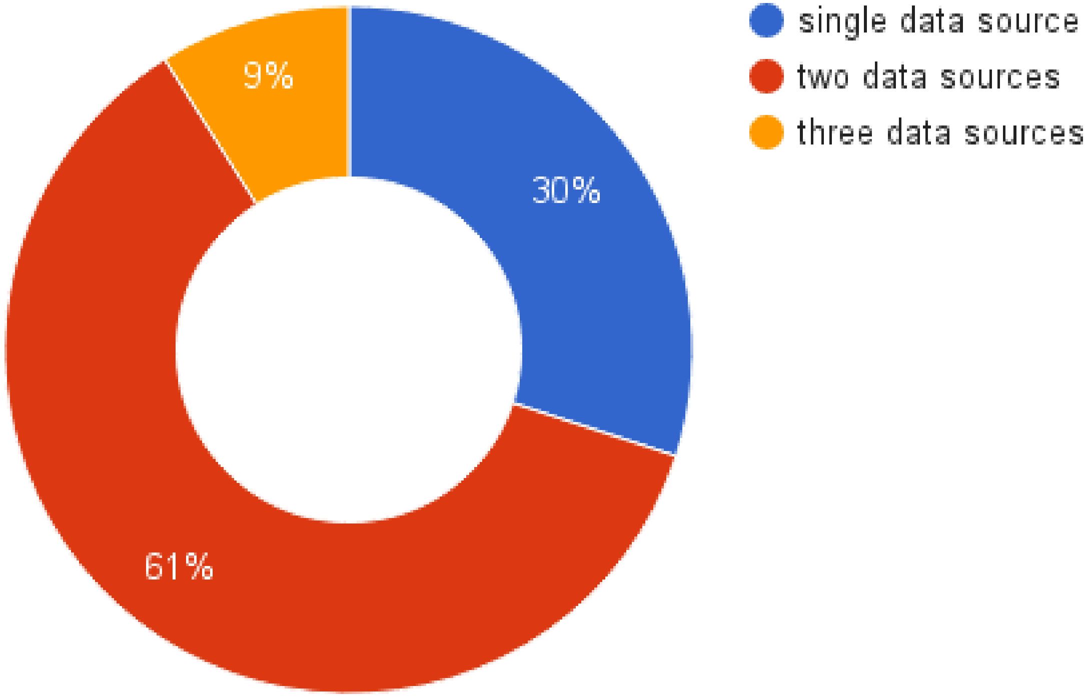
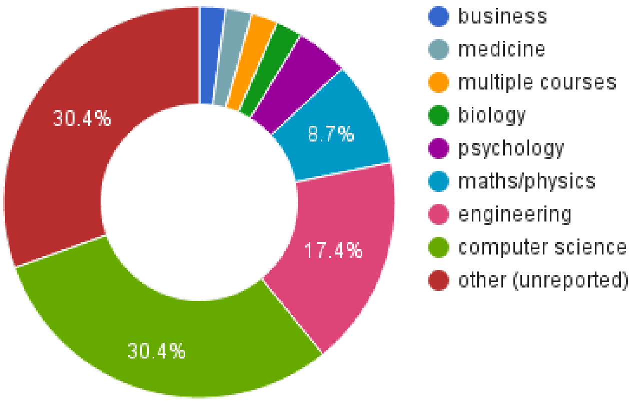


Figure 2.3: Percentage of Studies per Discipline

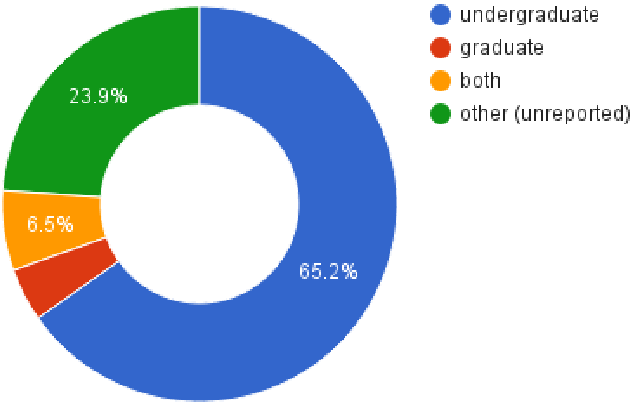


### 2.3.2 Course Levels and Disciplines

Of the 46 studies, the majority used data from courses administered as part of either computer science or engineering programmes at the undergraduate level. Of these, many focused on predicting performance of freshman and second year students enrolled in introductory level courses.

Reasons behind the lack of similar studies in other subject areas or at different course levels remain an assumption. Two partial explanations are provided. It may be that utilizing machine learning algorithms is an immediate advantage that researchers in the field of computer science have. It could also be that performance prediction has more impact when conducted at earlier years of college education. Figures 2.3 and 2.4 show the studies categorized according to disciplines and course levels.

Figure 2.4: Percentage of Studies per Course Level

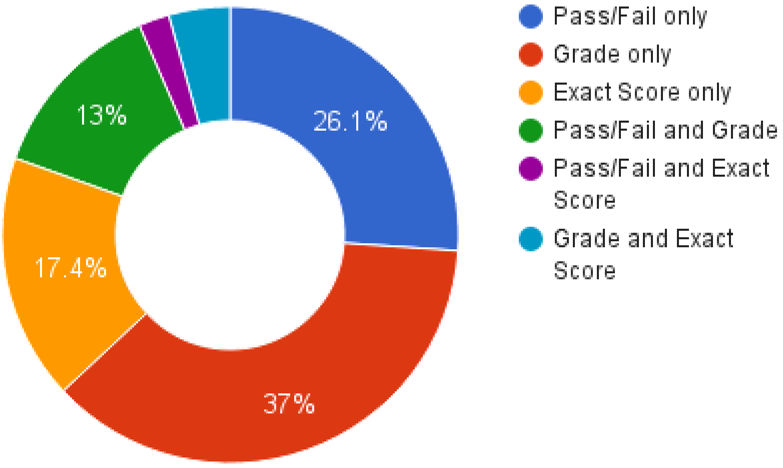


### 2.3.3 Granularity of Performance Prediction – Overall Success Versus Specific Outcome

The 46 studies also differed in how they interpreted performance. A generic approach is to predict pass or fail. Some of the studies that followed this approach include [66, 47, 71, 92, 97]. Others took a further step in predicting the classification of the degree or achievement [9, 74].

Figure 2.5 shows the studies grouped according to the outcomes they predicted. Some of these studies predicted overall success as well as specific outcomes. Predicting overall success in a timely manner serves as a mechanism of early intervention. Timely prediction of a range or scores or grades has, however, a more powerful formative value as it provides granular information about specific performance categories of students [8]. In recent years, prediction of student performance has become more fine-grained and sophisticated. Researchers now seek to predict actual scores for tests and assignments as well as final scores and grades for an

Figure 2.5: Percentage of Studies per Type of Prediction



entire course. In its simplest form, effective prediction of such outcomes has a two-part requirement. The type and amount of student data that is to be collected form one part and the data analysis and choice of prediction techniques form the other.

### 2.3.4 One-Off Versus Continuous Prediction – The Case of Summative and Formative Prediction Models

Performance prediction models provide a number of advantages for both students and teachers. In summative assessment environments, prediction only provides information about end-of-course performance or overall performance in later years of higher education.

In contrast, student-centred performance prediction aims to continuously provide information on student progress. For example, a prediction model integrated into an online learning environment could provide each student information about their predicted performance and how this performance relates to those of their peers. Such information is helpful to students because it helps them identify their strengths and weaknesses in certain topics. It also helps the teacher measure the overall progress of the class and identify those students that may require special supervision. Borrowing from assessment terminology, student performance prediction can therefore be categorized into formative and summative.

The parameters that are chosen to act as predictors of student performance essentially determine the type of prediction that is to be made. Some predictor attributes, once obtained, are highly unlikely to evolve over time. Hence, it is claimed that although they may provide important information about expected performance when used together with other dynamic variables, they hardly contribute to measuring student progress.

To demonstrate this, a scenario is considered where student success at the end of the first year of college is predicted. A common approach for building such prediction models is to train a machine learning algorithm with student performance data collected from the final years of high school and demographic background. For the purpose of predicting success at a critical point such as the end of the first year of college studies, it has been shown such data are indeed good predictors [44, 1, 36, 90, 63, 76, 19, 38].

However, much of the demographic data seldom changes and academic performance history from earlier years never does. For this reason, if a student's performance is predicted halfway through the first year of college using the same model, the predictions remain unchanged. Consequently, the use of such static parameters alone only allows making one-off predictions.

When combined with data that changes over time, demographic and previous academic performance data provide more information and can be used to measure progress. While static data provide information about a student's background, dynamic data such as the number of assignments completed to-date provide progress information. Together, static and dynamic data constitute a student profile in good models. There is no specific combination of static and dynamic student data that can apply to all prediction models. **Appendix A** shows the varying degrees of success obtained for the studies that considered both static and dynamic data.

One way of predicting progress is through predicting performance at specific intervals within a course. For instance, a study by Fernandez-Delgado et al. [29] showed how to predict performance on specific course modules. This is a good example of a formative predictive model. If students are informed in time about their expected performance on course modules, they have the opportunity to act accordingly.

In general, experiments that aim to predict student progress are most effective when predictions made at a specific point during a course utilize data from previously enrolled students up to the specific point of prediction. This can provide information on how students from previous cohorts with similar progress levels performed at the end of the course. In experiments that utilize machine learning algorithms, this implies training models with a portion of the available data set that covers the period from the start of the course to the specific point of prediction only. Such approach is discussed in Chapter 6.

Online educational platforms provide a unique advantage by collecting dynamic data and are the ideal platform for building progress prediction models on top of. Studies that utilize such platforms to provide predictions at several intervals of a course include [10, 58, 52]. None of these studies use demographic data of performance data from earlier levels of education.

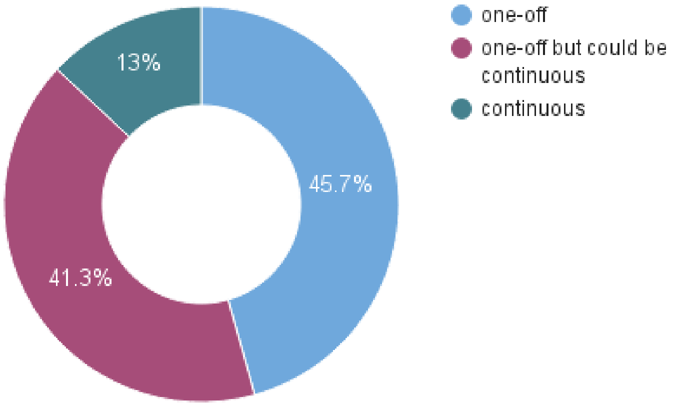
Because one-off predictions usually focus on predicting end-of-course performance, they hardly provide information about progress. In this manner, such predictions may be considered as having only summative value. Nonetheless, 19 of the 21 studies that made one-off predictions could be transformed to make continuous predictions because the prediction models could be applied to incomplete data obtained at several intervals during the course. The reason for not making predictions in this manner, perhaps, is due to the additional effort required to redesign experiments or to collect incomplete data several times during a course.

Figure 2.6 shows the studies categorized according to whether they make one-off (N=21) or continuous predictions (N=6) and those that make one-off predictions but could be transformed to provide continuous predictions (N=19).

### 2.3.5 What are Good Predictors of Student Success and How Good are the Predictions?

Despite the abundance of studies in performance prediction and the parameters used in making predictions, it is difficult to single out parameters as outstanding performance predictors. Two immediate reasons for this may be the variation in the setup of experiments and in the prediction algorithms used. Other factors include the amount of data and the course or discipline.

Figure 2.6: Percentage of Studies per Continuity of Predictions



Nonetheless, studies that used a higher number of predictor parameters and a larger number of student records reported better results. Because online learning and assessment environments simplify the collection of student activity data, and hence allow inclusion of parameters about such activity, studies that used online educational platforms reported significantly better results. 25 of the 46 studies used data from online learning platforms. Of these, 9 reported the number of records used to train their models and the accuracy of the models. Due to the varying number of students, accuracy could not be averaged directly. However, the average accuracy, weighted by the number of students, was 89%. Instead, for the 22 studies that did not use data from online learning platforms, the weighted accuracy was 80%.

Similarly, predictors built using data from a large cohort of students or utilized data collected over long periods of time, had high performance accuracy. The weighted accuracy for 9 studies that reported accuracy was again 89

It should, however, be noted that accuracy may not be the right choice of performance evaluation technique, especially when prediction is essentially a classification task. In such scenarios, other metrics such as precision, recall, F1 scores and False Positive Rates could be used. This is especially true when a prediction model is evaluated according to how many at-risk students it identifies and how many it fails to do so.

Forty-six percent of the studies used previous academic performance data, many in combination with other data such as online activity logs. Fourteen of the studies used demographic data in making predictions. A comparison of the data sources and the reported performance results revealed that demographic data are more effective when used in combination with two or more other data sources such as online activity logs and previous academic performance data. Indeed, those studies that used solely demographic data or in combination with only one other data source reported moderate performance levels.

In contrast, studies that included partial marks, mid-term results or assignment scores reported higher results. The highest results were reported by studies which included three or more of these parameters in their predictions. Details of the 46 studies and remarks on their adaptability to making continuous predictions are presented in **Appendix A**.

### 2.3.6 Prediction Techniques and Algorithms

The majority of the studies reviewed followed the approach of applying a range of machine learning algorithms to their data set and choosing the algorithm that reported the highest level of performance. The most common of these algorithms were Linear Regression, Neural Networks, Support Vector Machines, Naïve Bayes Classifier, and Decision Trees.

A few studies used a combination of classifiers for improved predictions [16, 9, 62].

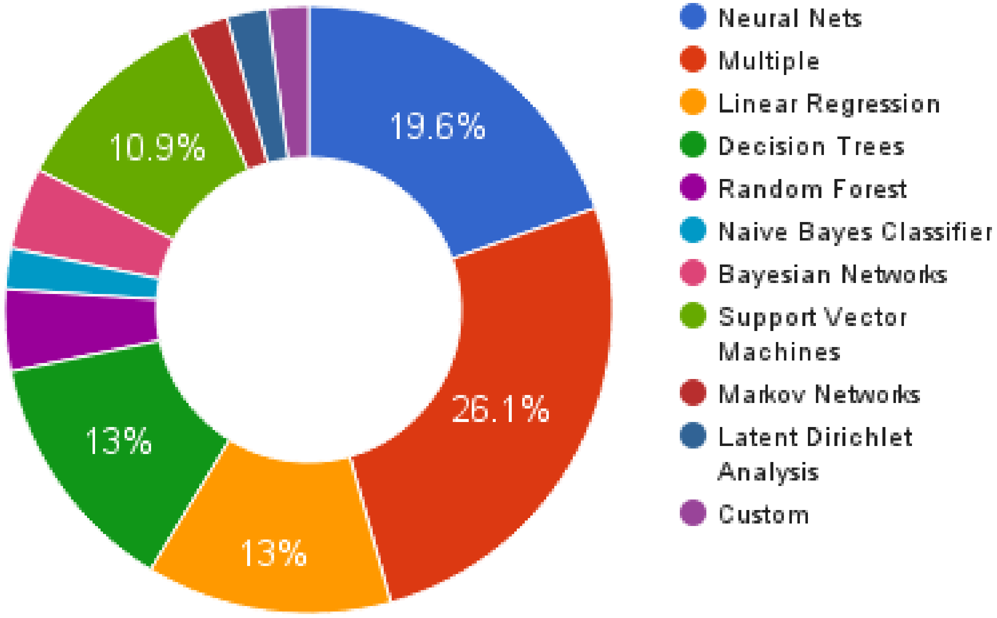
Other studies that followed less common approaches include those that used Markov Networks [82], Collaborative Multi-Regression models [24], smartphone data [93] and those that performed sentiment analysis of discussion forum posts in MOOCs [73].

Yet, some studies discussed algorithms developed for the sole purpose of student performance prediction [94, 61].

Despite the varied nature of the data used in the studies, no single algorithm can provide the best result in all prediction scenarios. However, 5 of the 9 studies that used a combination of algorithms reported over 90% prediction accuracy. Those that used Neural Networks, Random Forest, Decision Trees and Support Vector Machine algorithms reported over 80% accuracy.

Figure 2.7 shows the studies categorized by the type of prediction algorithm they used. Those that applied more than one algorithm are grouped under the 'multiple' category.

Figure 2.7: Percentage of Studies per Type of Prediction Algorithm used



### 2.3.7 Predictions in Massive Open Online Courses (MOOC)

One of the factors that may weaken the appeal of MOOCs as complementary to, or even long-term replacements for, traditional course administration practices is that they are plagued by high attrition rates (Jordan 2013). It is, hence, not surprising that most student performance prediction studies that involve MOOCs have focused on predicting student dropout.

Because MOOCs are inherently tied to online platforms, the task of collecting data regarding student activity is only as challenging as building the platforms themselves. MOOCs allow gathering immense amounts of data from students that do not necessarily attend courses from the same geographic region. Consequently, the diversity and size of the data collected by such platforms is unparalleled.

The impact of the unique advantage provided by using online course administration systems is immediately apparent in the performance levels of prediction models that are built on such data. Although the outcomes they predict are less complex when compared to models that predict grades or final scores, predictors in MOOCs consistently perform better than their counterparts built on top of traditional educational settings. One possible reason for this high performance is the ability of such platforms to capture data about student traits and activities that are only expressed in online learning environments. These may include participation in discussion forums and amount of time spent on completing tasks. Another reason is that the amount of data used to build the prediction models is significantly large and leads to improved prediction.

Many studies also set out to explore pedagogical or administrative factors that affect student success in MOOCs. Among such factors are ownership and length of posts in online discussion forums [99].

Other activities that may be used as features for building MOOC dropout prediction models include number of video lecture downloads, number of completed quizzes, number of completed tasks, click-stream data, the amount of time spent on course modules and the number of days students are active [100, 74, 58, 10, 79, 86, 20, 49].

### 2.3.8 The Potential Paradox of an Effective Performance Prediction System

A straightforward approach to testing the performance of models that predict favorable outcomes is to compare such predictions with actual outcomes. Such an approach, however, may not apply to systems that predict unfavorable outcomes as the effectiveness of these predictive systems is rooted in their ability to provide information that would help avert such outcomes.

If common evaluation techniques are used to measure the performance of prediction systems that help avert unfavorable outcomes, evaluation results will lead to the conclusion that these prediction models do not perform well. By the same argument, a conclusion that these models are good predictors implies bearing the consequences of unfavorable outcomes. Although this paradoxical nature of evaluating prediction models is somewhat nonexistent in models that predict natural catastrophes, it is still evident in models that provide timely prediction of student performance.

Concretely, a question is raised about how to statistically evaluate the performance of a model that predicts an outcome in a timely manner when the true performance of the model is measured in its ability to help avoid that same outcome. In the case of student performance prediction, timely prediction helps identify students who are at risk of failing. If that information is acted upon properly, the undesired outcome, failing a course, is averted. Classical evaluation techniques such as prediction accuracy would provide the misleading conclusion that the model does not perform well, when it actually does so.

The performance of such non-traditional prediction models can be tested in two ways. An approach that is common to many well-designed research experiments is to divide subjects into experimental and control groups. In a class of students divided into such two groups, continuous and timely prediction will be provided to both groups but only those in the experimental group will be supervised according to the information provided by the prediction model. At the end of the course, analyzing how many at-risk students in both groups, as predicted by the model, improved and eventually passed the course shall reveal the true performance of the prediction model. Traditional performance evaluation techniques could be used to evaluate the performance of the model on the control group. Statistical methods that measure how varied two sets of outcomes are, can be used to judge how good the prediction model performs on the experimental group. Large differences between predictions and actual outcomes for the experimental group would then imply high performance of the prediction model.

This approach, however, entails more risk for students in the control group as they may not benefit from the prediction and may indeed fail the course. This could explain why none of the 46 studies adopted this or a similar controlled experiment approach.

An empirical approach that does not partition students into distinct groups involves periodic evaluation of only those students that the prediction model deems to be at-risk of failing. Although this method implies more work on part of the teacher, it provides invaluable information about the actual progress of the student. As noted in earlier sections, this approach is typical in formative assessment scenarios.

### 2.3.9 Summary

The majority of performance prediction research has focused on the disciplines of computer science and engineering. Although not far-fetched, the argument that researchers in these disciplines have at their disposal tools and know-how to build better prediction models may not explain the observation very well.

Although earlier research sought to establish whether it was possible to make predictions that were binary in nature, recent studies have managed to predict specific grades with high accuracy. In fact, some studies go as far as predicting specific scores. In general, predicting outcome becomes more difficult as the number of possible outcomes grows. Hence, studies that predict actual scores are usually less accurate than those that predict pass or fail. However, significant increase in performance is noted in score prediction models with increase in the amount of data.

Increase in training data size does not necessarily imply improved prediction accuracy. It is possible that after a certain point, prediction performance may not improve despite the introduction of new data. This is because prediction parameters and algorithms are just as important. After all, it is learning algorithms that use a large number of parameters – algorithms with low bias – and large training data, which provide low variance, that constitute a good prediction model.

Whether prediction is intended for summative or formative purposes highly determines the approach to building the model. While educational policy makers may be more interested in determining whether national and standardized tests are good predictors of long-term student performance, teachers benefit more from information that provides insight into student progress. Students benefit from timely information about their progress because it gives them the opportunity to act accordingly. The parameters chosen to build prediction models should reflect this notion of progress as well.

## 2.4 Conclusion

Online peer-assessment data provide rich information about students and may be used to build performance prediction models. None of the studies reviewed, however, used data from peer-assessment environments.

Indeed, an extensive search of educational conference and journal databases such as ACM, IEEExplore, ERIC and Google Scholar did not result in any study that used peer-assessment data for building prediction models.

Although a similar approach is used to evaluate the importance of the web-based peer-assessment system discussed in the next chapter, the chapters that follow up argue that, based on statistical analysis and utilisation of the data extracted from the system, its perceived advantages are validated and corroborated.

Chapters 5 and 6 explore how peer-assessment data can be utilized to build models that can predict end-of-course performance as well as track student progress.

# 3

# Peer-Assessment for Promoting Student Engagement

## 3.1 Introduction

One of the challenges of both traditional and contemporary instructional media in higher education is creating a sustainable teaching-learning environment that ensures continuous engagement of students and provides efficient means of assessing their performance [15]. Summative assessment in campus-based classrooms with hundreds of students is seldom carried out enough throughout a course to identify those students who might be at risk of failing or dropping out. The MOOC phenomenon, although it promises to reach out to and educate more students across the globe, has suffered the same fate, if not worse. Attrition rates in MOOCs are astoundingly high [46]. It may be argued, given the hundreds of thousands of students that enroll in a MOOC, even a small percentage of completions implies a high number of students completing the course with success when compared to actual numbers in campus-based courses. This argument, however, does little to address the issue of student engagement.

Is it because of cognitive challenges that they face that many students drop out of courses? Is it why students tend to wait until the final weeks of a course to prepare for their final exam? Or, is it because of passive participation that does not elicit their engagement? Whichever the reason, is it possible to develop a mechanism that can identify in a timely manner students who are at risk of dropping out or failing? It was believed that the attempt to discover the true reason behind student dropouts and course failures required the use of efficient methodologies and technological tools. It was also believed that the initial attempt of implementing these methodologies and tools should not do away with traditional instruction methods but augment them in a way that would solicit sustained student engagement. With these beliefs, a web-based peer-assessment tool was developed and utilized in several editions of three courses spanning four years. This chapter is devoted to the discussion of this web-based peer-assessment platform.

There are several variations of peer-assessment methodologies, which have been integrated into traditional classrooms. PeerGrader [33], PRAISE [22], PeerWise [23] and PeerScholar [70] are the most notable for their formative nature and the way they involve students in several assessment stages. Similarly, it was decided to involve students in a question-answer-evaluation loop that would include some game elements and in which students themselves would be the lead actors. The hypothesis was that being actively involved in such activities would encourage them to keep up with the pace of the course and to study regularly in a deeper, more engaging manner.

## 3.2 Design of Peer-Assessment Activities

The peer-assessment activities involved cycles of three tasks in which students were required to submit questions about topics discussed in class, respond to a subset of the questions that were submitted and rate the responses of their peers to those questions.

It was decided not to make participation in PA activities mandatory. This was because it was believed mandatory participation would mask true effects of the proposed approach and would force it to somewhat resemble summative assessment. However, students were rewarded according to the extent of their involvement and the number of performance points they collected throughout these activities.

Every week for the duration the course, students were required to ask a question regarding topics covered during the most recent lectures. Questions varied from typical assessment queries to requests for clarification and to inquiries that would require deeper insight into a theme. When submitting their task, students tagged their question using at least two keywords.

Next, each question was assigned to five students at random. Students then rated the quality of the questions they were assigned on a scale of 1 to 5 across three dimensions – interestingness, difficulty and relevance.

Once students submitted their questions, the teacher would select a subset of the questions, taking into account their difficulty levels, relevance and interestingness.

Questions would then be automatically distributed to students in a random manner. The task distribution module of the peer-assessment platform ensured that each question was assigned to at least four students.

In rating responses of their peers to a question, students would be assigned a specific number of points, computed using the number of answers submitted to that question. For the purpose of the peer-assessment activities, these points were referred to as coins. Students would then distribute the coins over the responses, where each response could be assigned a range of coins between zero and five, inclusive.

After a cycle of tasks was complete, statistics about responses were made available for students through their profile pages. In order to account for task incompletions, which occurred mainly because participation was not mandatory, the number of points earned for answers was reported in terms of the number of Effective Coins (EC). This is the ratio of the number of coins earned for an answer to a question to the total number of coins earned for all answers to that question. High EC values for an answer implied that the answer was found by raters to be superior to other answers to the same question and low EC values indicated otherwise.

In addition to EC values, statistics such as the number of completed tasks, the number of tasks for which deadlines were missed, the number of outstanding tasks and class standings in terms of weekly percentiles were made available to students.

## 3.3 Implementation of the Framework

The peer-assessment framework was implemented as a Java web application using the Struts2 MVC framework. The front-end was designed using Java Server Pages (JSP) while the backend utilized Hibernate ORM for high-level mapping of objects to MySQL database relations.

In order to guarantee that the system was used only by students at the University of Trento, it was integrated with the university's authentication infrastructure.

The system was composed of four main modules - The instructor module, the student module, the question selection module and the Q&A summary module.

### 3.3.1 The instructor module

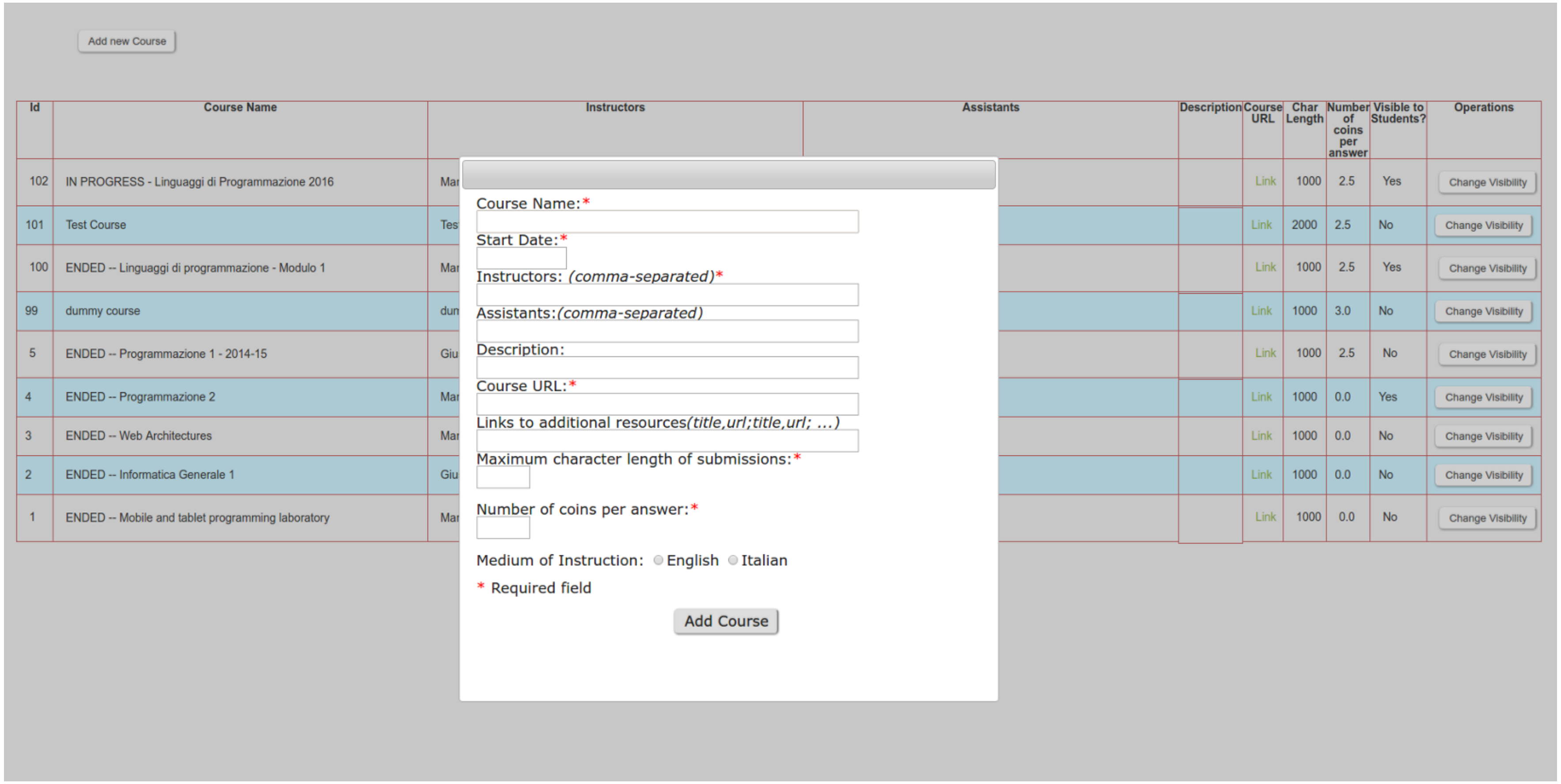
The instructor module allows the instructor to add new courses into the system, add lectures for that course and assign new tasks to all students. The module also allows the teacher access to the complete list of students and their activities in the system. Using the module, the teacher could contact students.

The system automatically sends an email to students when a new task is available. The teacher has the option to add notes and instructions to the email.

The instructor module can also generate several types of reports about student activities. It also lets the teacher download questions and Q&A sets for each lecture.

In courses where teaching assistants are available, they may be employed to rate student answers. Figures 3.1 to 3.7 provides snapshots of this module.

Figure 3.1: Peer-Assessment System - Courses Dashboard



Add new Course

| Id | Course Name | Instructors | Assistants | Description | Course URL | Char Length | Number of coins per answer | Visible to Students? | Operations |
|---|---|---|---|---|---|---|---|---|---|
| 102 | IN PROGRESS - Linguaggi di Programmazione 2016 | Mar |  |  | Link | 1000 | 2.5 | Yes | Change Visibility |
| 101 | Test Course | Tes |  |  | Link | 2000 | 2.5 | No | Change Visibility |
| 100 | ENDED -- Linguaggi di programmazione - Modulo 1 | Mar |  |  | Link | 1000 | 2.5 | Yes | Change Visibility |
| 99 | dummy course | dun |  |  | Link | 1000 | 3.0 | No | Change Visibility |
| 5 | ENDED -- Programmazione 1 - 2014-15 | Giu |  |  | Link | 1000 | 2.5 | No | Change Visibility |
| 4 | ENDED -- Programmazione 2 | Mar |  |  | Link | 1000 | 0.0 | Yes | Change Visibility |
| 3 | ENDED -- Web Architectures | Mar |  |  | Link | 1000 | 0.0 | No | Change Visibility |
| 2 | ENDED -- Informatica Generale 1 | Giu |  |  | Link | 1000 | 0.0 | No | Change Visibility |
| 1 | ENDED -- Mobile and tablet programming laboratory | Mar |  |  | Link | 1000 | 0.0 | No | Change Visibility |

Course Name:*

Start Date:*

Instructors: *(comma-separated)**

Assistants:*(comma-separated)*

Description:

Course URL:*

Links to additional resources*(title,url;title,url; ...)*

Maximum character length of submissions:*

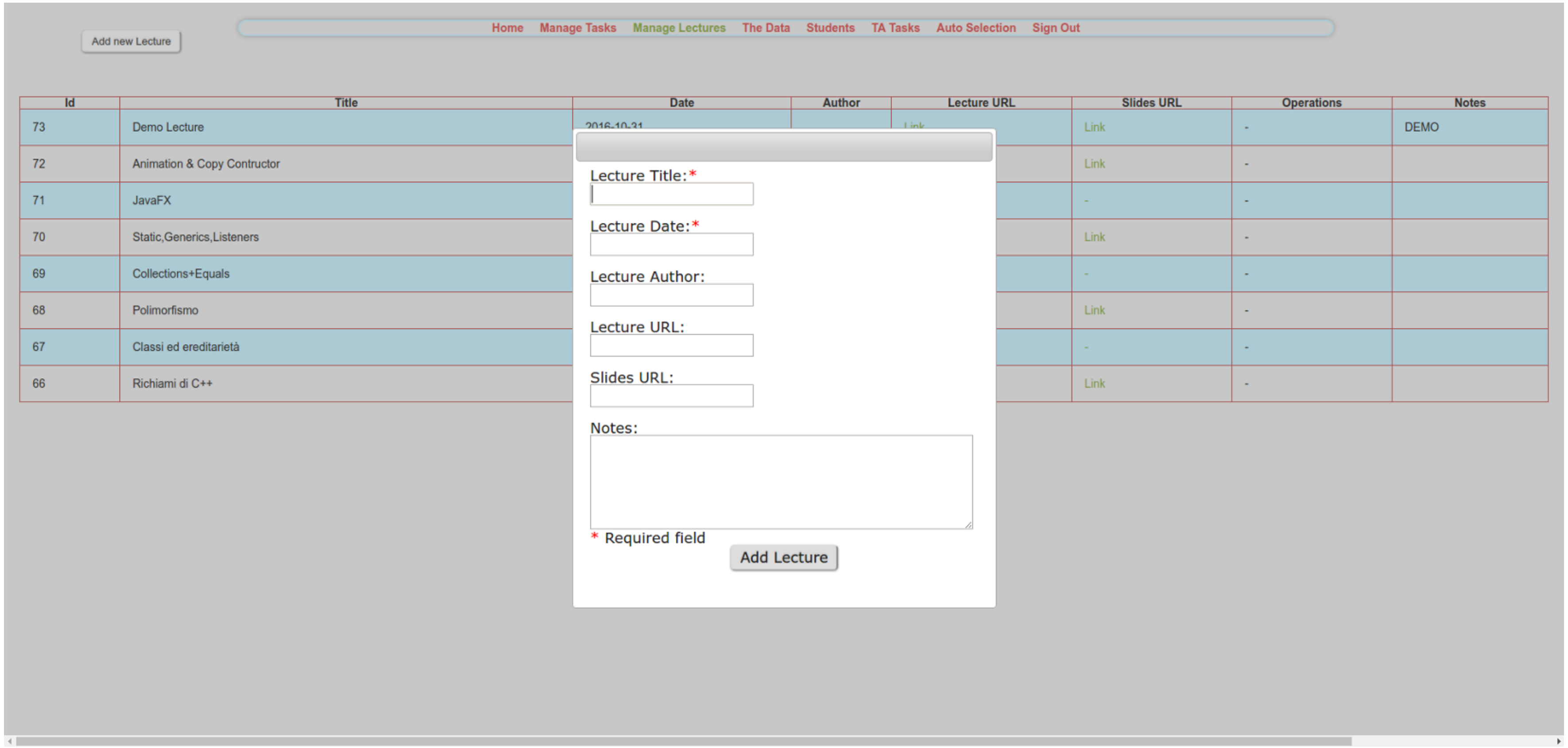
Number of coins per answer:*

Medium of Instruction: ○ English ○ Italian

* Required field

Add Course

Figure 3.2: Peer-Assessment System - Lectures Dashboard



| Id | Title | Date | Author | Lecture URL | Slides URL | Operations | Notes |
|---|---|---|---|---|---|---|---|
| 73 | Demo Lecture | 2016-10-31 | | Link | Link | - | DEMO |
| 72 | Animation & Copy Contructor | | | | Link | - | |
| 71 | JavaFX | | | | - | - | |
| 70 | Static,Generics,Listeners | | | | Link | - | |
| 69 | Collections+Equals | | | | - | - | |
| 68 | Polimorfismo | | | | Link | - | |
| 67 | Classi ed ereditarietà | | | | - | - | |
| 66 | Richiami di C++ | | | | Link | - | |

Lecture Title:*

Lecture Date:*

Lecture Author:

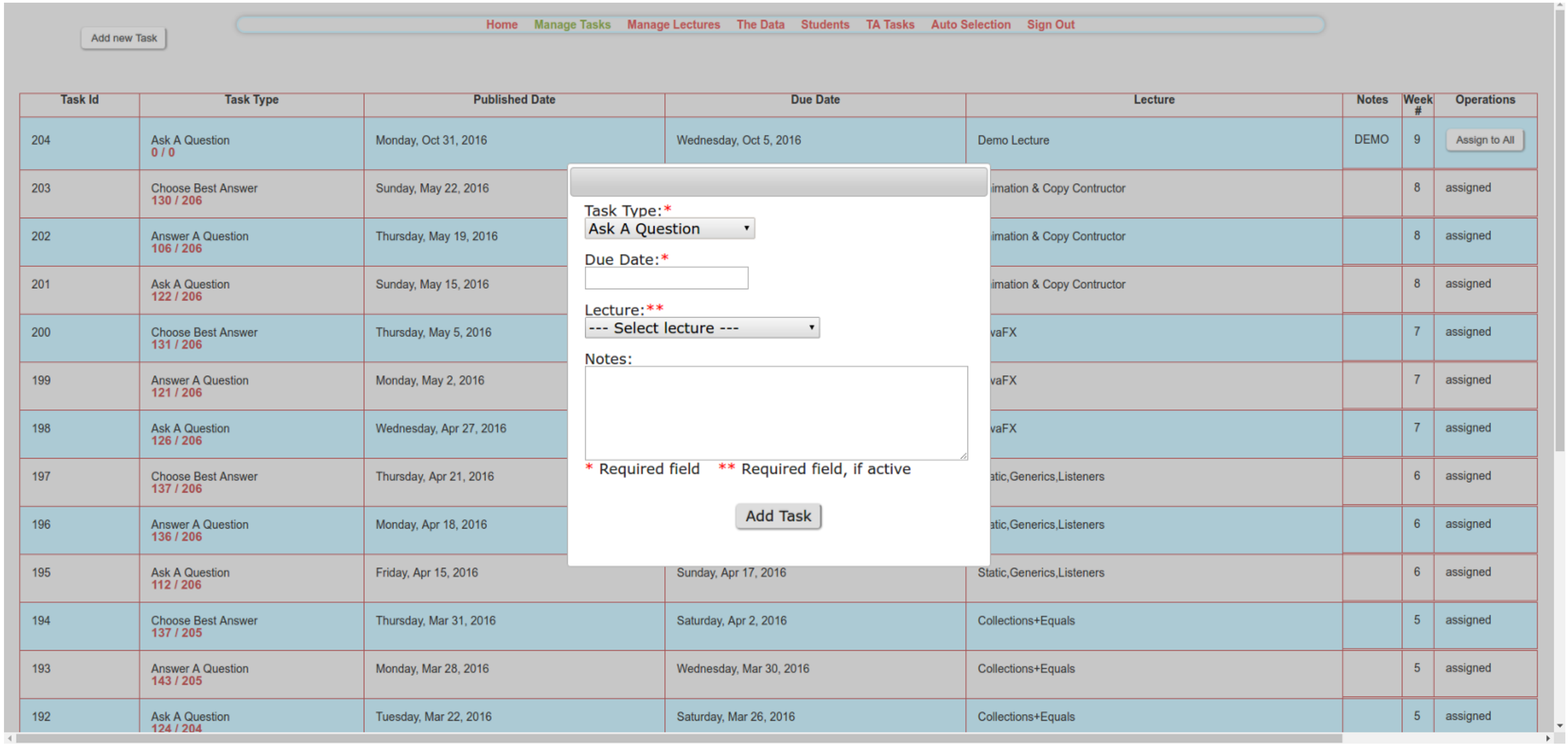
Lecture URL:

Slides URL:

Notes:

* Required field

Figure 3.3: Peer-Assessment System - Tasks Dashboard



| Task Id | Task Type | Published Date | Due Date | Lecture | Notes | Week # | Operations |
|---|---|---|---|---|---|---|---|
| 204 | Ask A Question<br>0 / 0 | Monday, Oct 31, 2016 | Wednesday, Oct 5, 2016 | Demo Lecture | DEMO | 9 | Assign to All |
| 203 | Choose Best Answer<br>130 / 206 | Sunday, May 22, 2016 | | imation & Copy Contructor | | 8 | assigned |
| 202 | Answer A Question<br>106 / 206 | Thursday, May 19, 2016 | | imation & Copy Contructor | | 8 | assigned |
| 201 | Ask A Question<br>122 / 206 | Sunday, May 15, 2016 | | imation & Copy Contructor | | 8 | assigned |
| 200 | Choose Best Answer<br>131 / 206 | Thursday, May 5, 2016 | | vaFX | | 7 | assigned |
| 199 | Answer A Question<br>121 / 206 | Monday, May 2, 2016 | | vaFX | | 7 | assigned |
| 198 | Ask A Question<br>126 / 206 | Wednesday, Apr 27, 2016 | | vaFX | | 7 | assigned |
| 197 | Choose Best Answer<br>137 / 206 | Thursday, Apr 21, 2016 | | atic,Generics,Listeners | | 6 | assigned |
| 196 | Answer A Question<br>136 / 206 | Monday, Apr 18, 2016 | | atic,Generics,Listeners | | 6 | assigned |
| 195 | Ask A Question<br>112 / 206 | Friday, Apr 15, 2016 | Sunday, Apr 17, 2016 | Static,Generics,Listeners | | 6 | assigned |
| 194 | Choose Best Answer<br>137 / 205 | Thursday, Mar 31, 2016 | Saturday, Apr 2, 2016 | Collections+Equals | | 5 | assigned |
| 193 | Answer A Question<br>143 / 205 | Monday, Mar 28, 2016 | Wednesday, Mar 30, 2016 | Collections+Equals | | 5 | assigned |
| 192 | Ask A Question<br>124 / 204 | Tuesday, Mar 22, 2016 | Saturday, Mar 26, 2016 | Collections+Equals | | 5 | assigned |

Task Type:*
Ask A Question

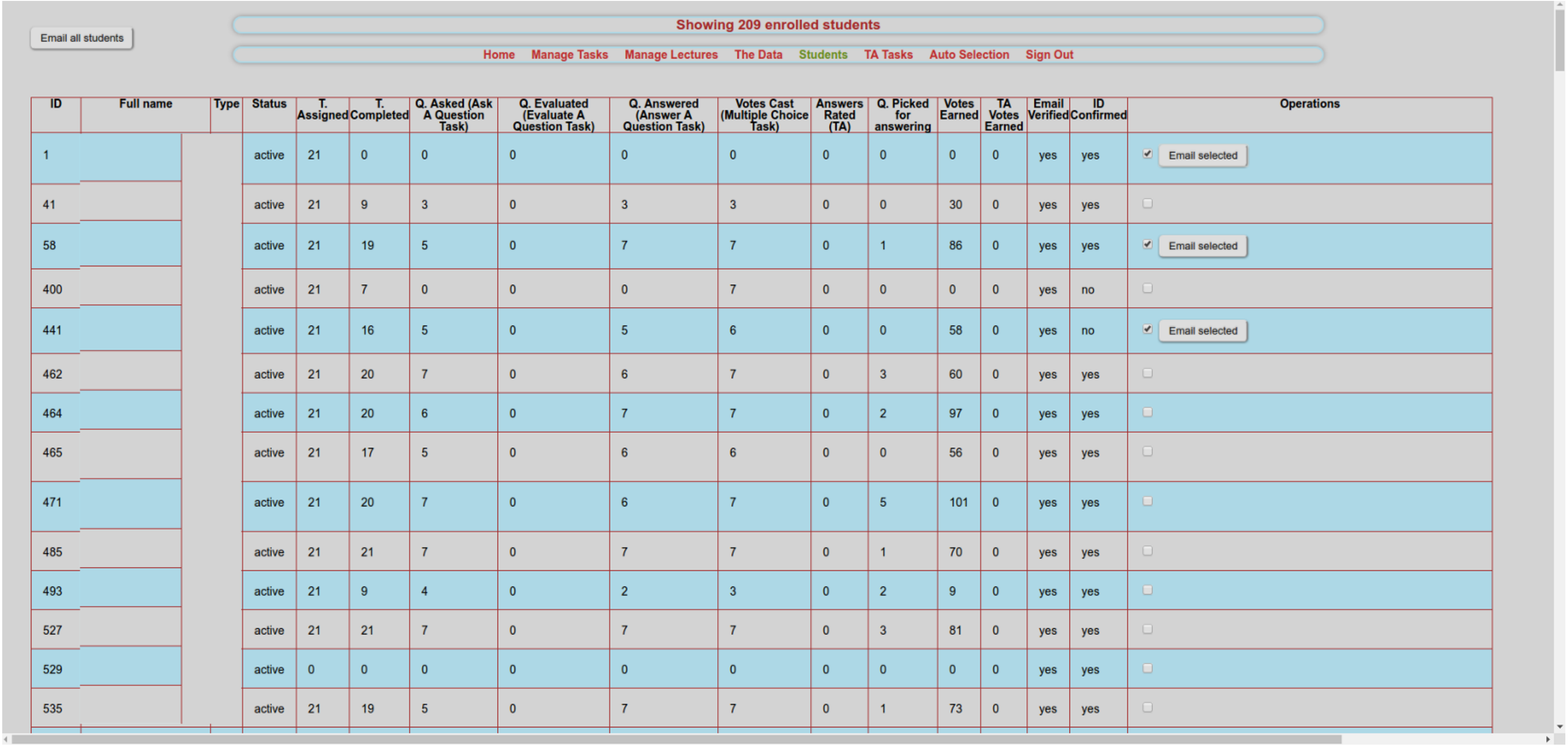
Due Date:*

Lecture:**
--- Select lecture ---

Notes:

* Required field   ** Required field, if active

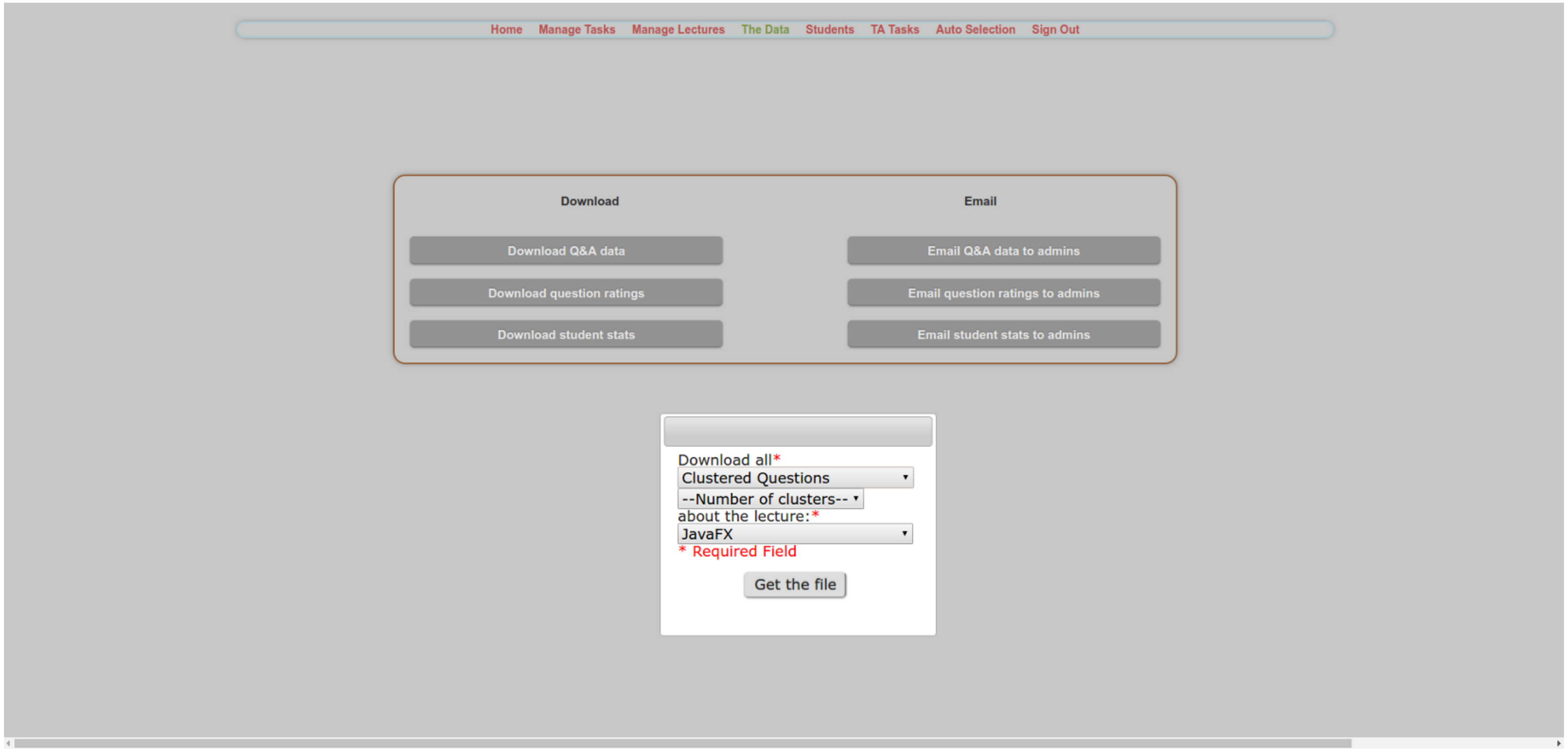
Figure 3.4: Peer-Assessment System - Students Dashboard



Showing 209 enrolled students

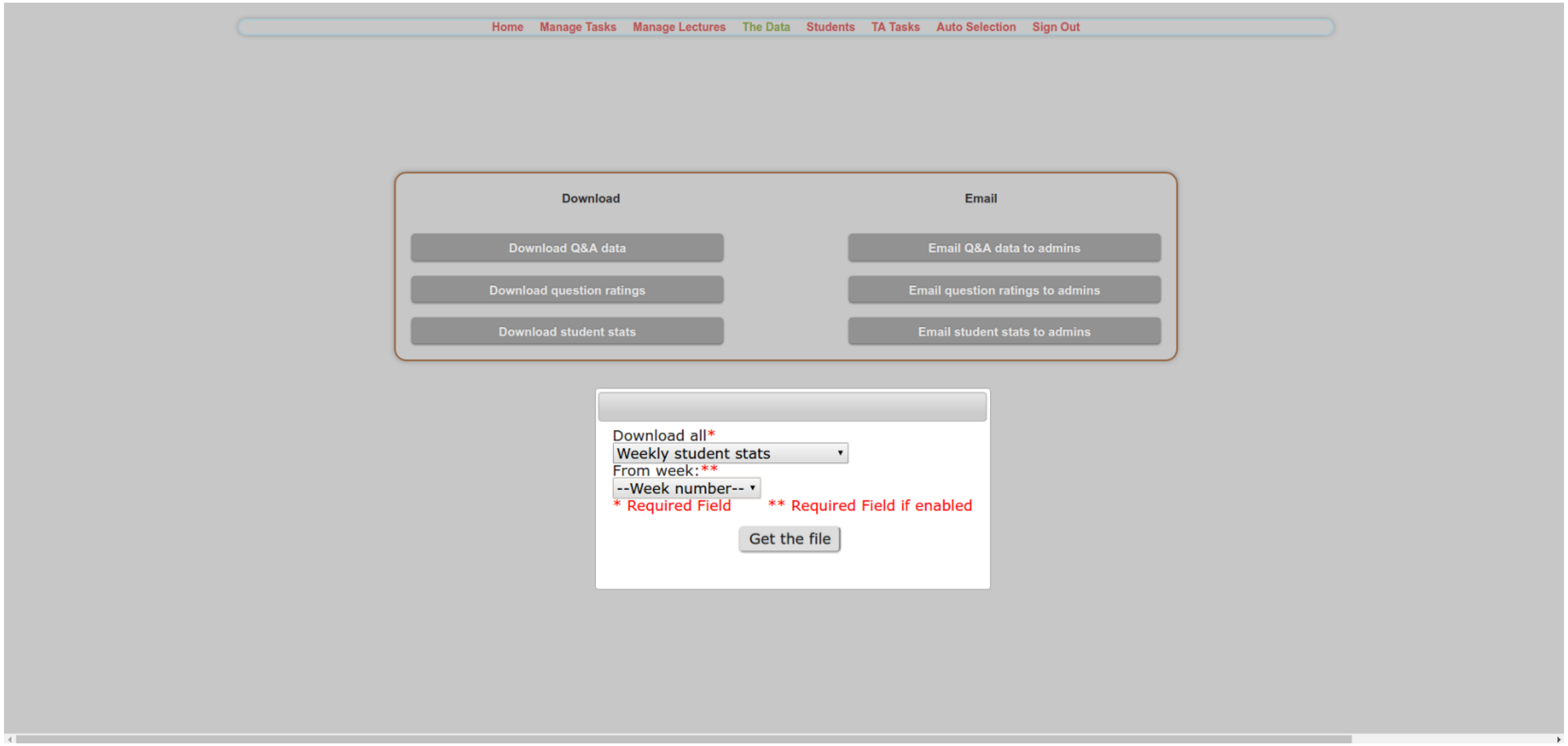
| ID | Full name | Type | Status | T. Assigned | T. Completed | Q. Asked (Ask A Question Task) | Q. Evaluated (Evaluate A Question Task) | Q. Answered (Answer A Question Task) | Votes Cast (Multiple Choice Task) | Answers Rated (TA) | Q. Picked for answering | Votes Earned | TA Votes Earned | Email Verified | ID Confirmed | Operations |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | active | 21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | yes | yes | ☑ Email selected |
| 41 | | | active | 21 | 9 | 3 | 0 | 3 | 3 | 0 | 0 | 30 | 0 | yes | yes | ☐ |
| 58 | | | active | 21 | 19 | 5 | 0 | 7 | 7 | 0 | 1 | 86 | 0 | yes | yes | ☑ Email selected |
| 400 | | | active | 21 | 7 | 0 | 0 | 0 | 7 | 0 | 0 | 0 | 0 | yes | no | ☐ |
| 441 | | | active | 21 | 16 | 5 | 0 | 5 | 6 | 0 | 0 | 58 | 0 | yes | no | ☑ Email selected |
| 462 | | | active | 21 | 20 | 7 | 0 | 6 | 7 | 0 | 3 | 60 | 0 | yes | yes | ☐ |
| 464 | | | active | 21 | 20 | 6 | 0 | 7 | 7 | 0 | 2 | 97 | 0 | yes | yes | ☐ |
| 465 | | | active | 21 | 17 | 5 | 0 | 6 | 6 | 0 | 0 | 56 | 0 | yes | yes | ☐ |
| 471 | | | active | 21 | 20 | 7 | 0 | 6 | 7 | 0 | 5 | 101 | 0 | yes | yes | ☐ |
| 485 | | | active | 21 | 21 | 7 | 0 | 7 | 7 | 0 | 1 | 70 | 0 | yes | yes | ☐ |
| 493 | | | active | 21 | 9 | 4 | 0 | 2 | 3 | 0 | 2 | 9 | 0 | yes | yes | ☐ |
| 527 | | | active | 21 | 21 | 7 | 0 | 7 | 7 | 0 | 3 | 81 | 0 | yes | yes | ☐ |
| 529 | | | active | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | yes | yes | ☐ |
| 535 | | | active | 21 | 19 | 5 | 0 | 7 | 7 | 0 | 1 | 73 | 0 | yes | yes | ☐ |

Figure 3.5: Peer-Assessment System - Data Export Dashboard 1



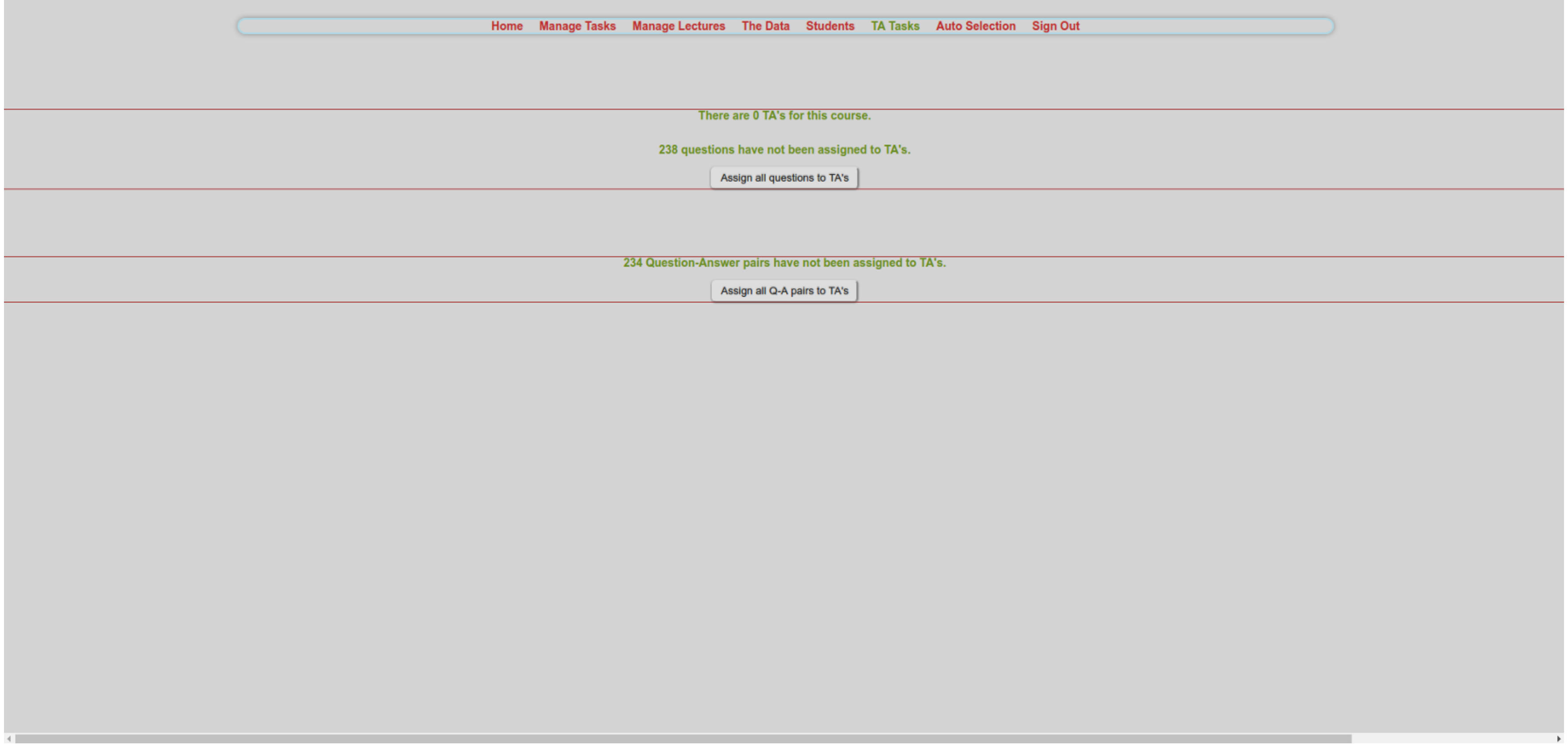
Download all*
Clustered Questions
--Number of clusters--
about the lecture:*
JavaFX
* Required Field

Figure 3.6: Peer-Assessment System - Data Export Dashboard 2



Download all*
Weekly student stats
From week:**
--Week number--
* Required Field ** Required Field if enabled

Figure 3.7: Peer-Assessment System - Teaching Assistant Dashboard



There are 0 TA's for this course.

238 questions have not been assigned to TA's.

234 Question-Answer pairs have not been assigned to TA's.

### 3.3.2 The Student Module

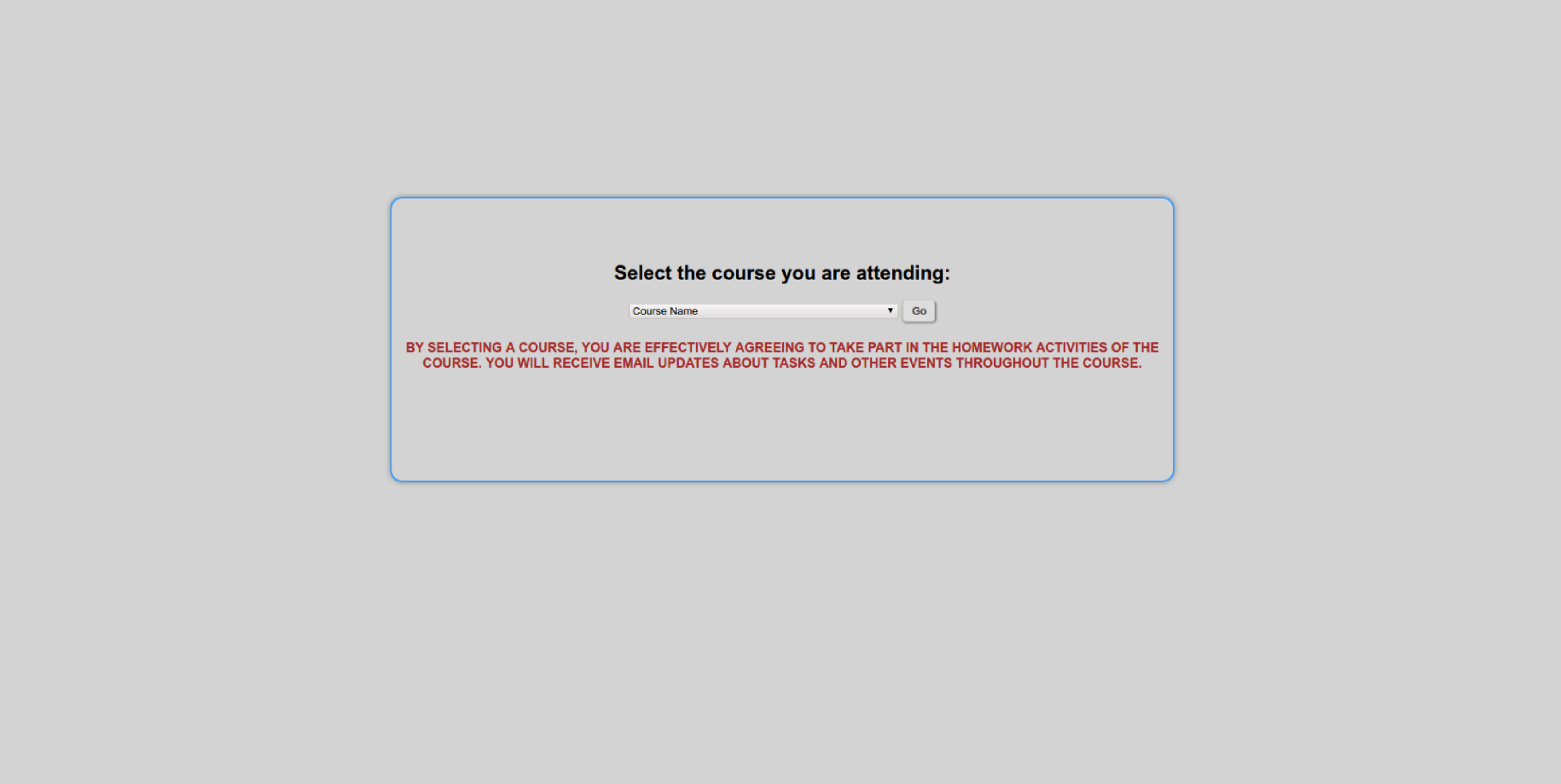
The student module of the framework was designed to provide students access to tasks and other information. Students have to sign in using their university credentials. After successful login, they are presented with a menu of courses to choose from, as the system can be used with several courses at once.

After selecting a course, students are directed to their homepage. The homepage provides a list of tasks that are not completed and a history bar with recent activities. It also provides a sidebar with a summary of activities such as the number of tasks completed, the total number of tasks assigned, the total number of points earned and the leading number of points for the class.

Students also have access to a statistics page, which provides visual information about their activities. This information is presented as the task completion ratio, for each type of task, and weekly standings in terms of percentiles of points earned for a single week. The lowest and the highest percentiles in the class for each week are also reported.

Figures 3.8 to 3.13 provide snapshots of this module.

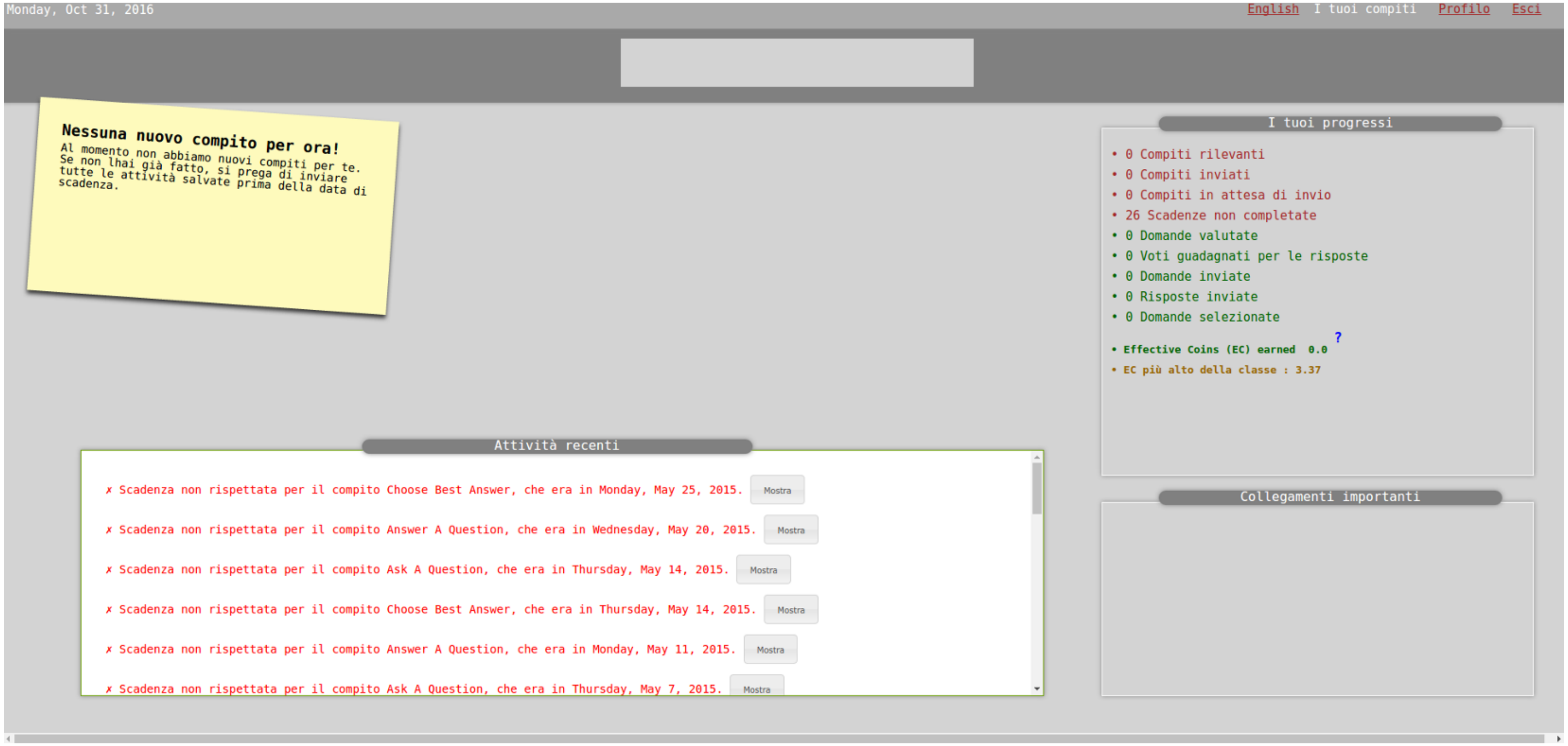
Figure 3.8: Peer-Assessment System - Course Enrolment Page



Select the course you are attending:

BY SELECTING A COURSE, YOU ARE EFFECTIVELY AGREEING TO TAKE PART IN THE HOMEWORK ACTIVITIES OF THE COURSE. YOU WILL RECEIVE EMAIL UPDATES ABOUT TASKS AND OTHER EVENTS THROUGHOUT THE COURSE.

Figure 3.9: Peer-Assessment System - Student Homepage 1



Monday, Oct 31, 2016

**Nessuna nuovo compito per ora!**

Al momento non abbiamo nuovi compiti per te. Se non lhai già fatto, si prega di inviare tutte le attività salvate prima della data di scadenza.

I tuoi progressi

- 0 Compiti rilevanti
- 0 Compiti inviati
- 0 Compiti in attesa di invio
- 26 Scadenze non completate
- 0 Domande valutate
- 0 Voti guadagnati per le risposte
- 0 Domande inviate
- 0 Risposte inviate
- 0 Domande selezionate

?

- **Effective Coins (EC) earned 0.0**
- **EC più alto della classe : 3.37**

Attività recenti

✗ Scadenza non rispettata per il compito Choose Best Answer, che era in Monday, May 25, 2015. Mostra

✗ Scadenza non rispettata per il compito Answer A Question, che era in Wednesday, May 20, 2015. Mostra

✗ Scadenza non rispettata per il compito Ask A Question, che era in Thursday, May 14, 2015. Mostra

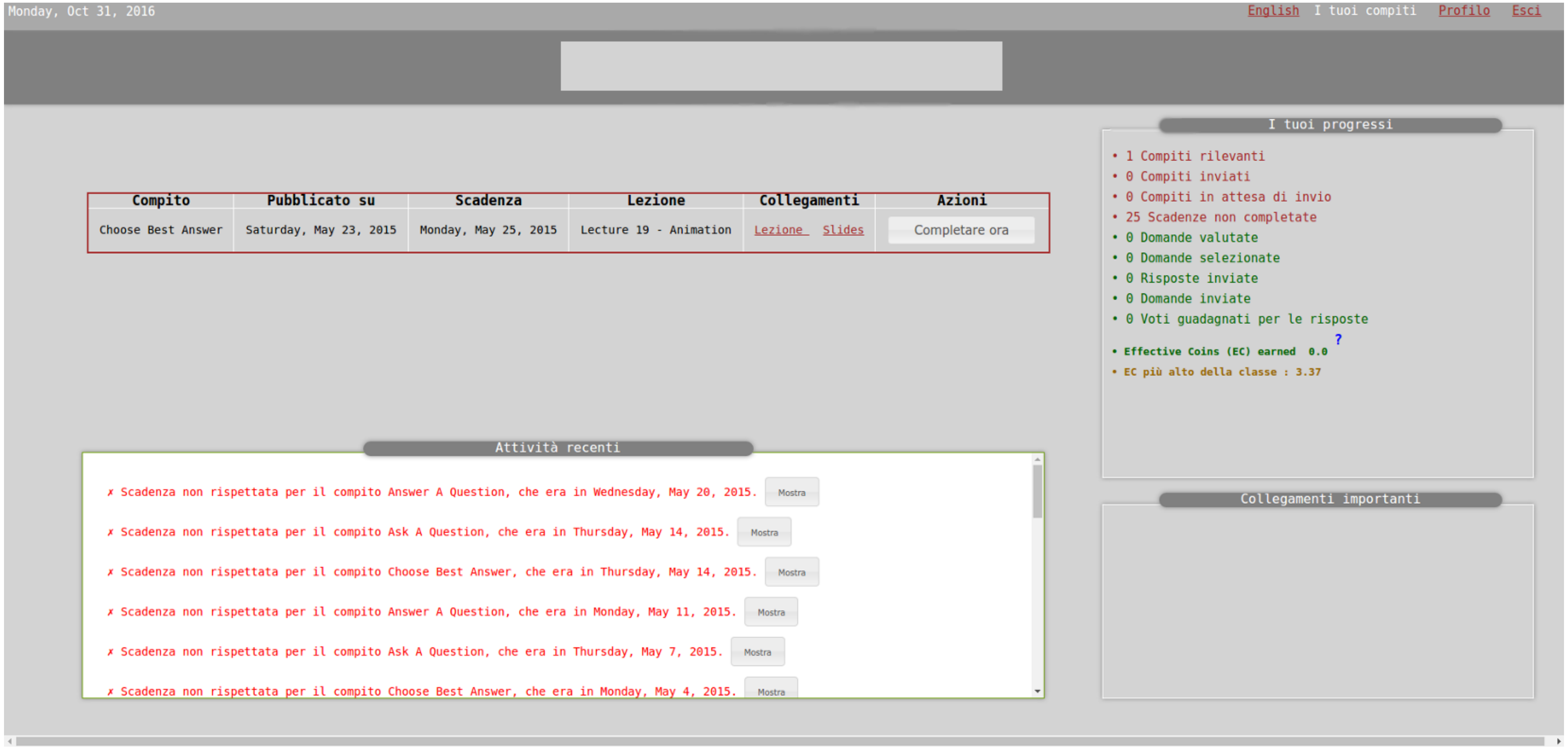
✗ Scadenza non rispettata per il compito Choose Best Answer, che era in Thursday, May 14, 2015. Mostra

✗ Scadenza non rispettata per il compito Answer A Question, che era in Monday, May 11, 2015. Mostra

✗ Scadenza non rispettata per il compito Ask A Question, che era in Thursday, May 7, 2015. Mostra

Collegamenti importanti

Figure 3.10: Peer-Assessment System - Student Homepage 2



Monday, Oct 31, 2016

| Compito | Pubblicato su | Scadenza | Lezione | Collegamenti | Azioni |
|---|---|---|---|---|---|
| Choose Best Answer | Saturday, May 23, 2015 | Monday, May 25, 2015 | Lecture 19 - Animation | Lezione Slides | Completare ora |

I tuoi progressi

- 1 Compiti rilevanti
- 0 Compiti inviati
- 0 Compiti in attesa di invio
- 25 Scadenze non completate
- 0 Domande valutate
- 0 Domande selezionate
- 0 Risposte inviate
- 0 Domande inviate
- 0 Voti guadagnati per le risposte
- **Effective Coins (EC) earned 0.0** ?
- **EC più alto della classe : 3.37**

Attività recenti

✗ Scadenza non rispettata per il compito Answer A Question, che era in Wednesday, May 20, 2015. Mostra

✗ Scadenza non rispettata per il compito Ask A Question, che era in Thursday, May 14, 2015. Mostra

✗ Scadenza non rispettata per il compito Choose Best Answer, che era in Thursday, May 14, 2015. Mostra

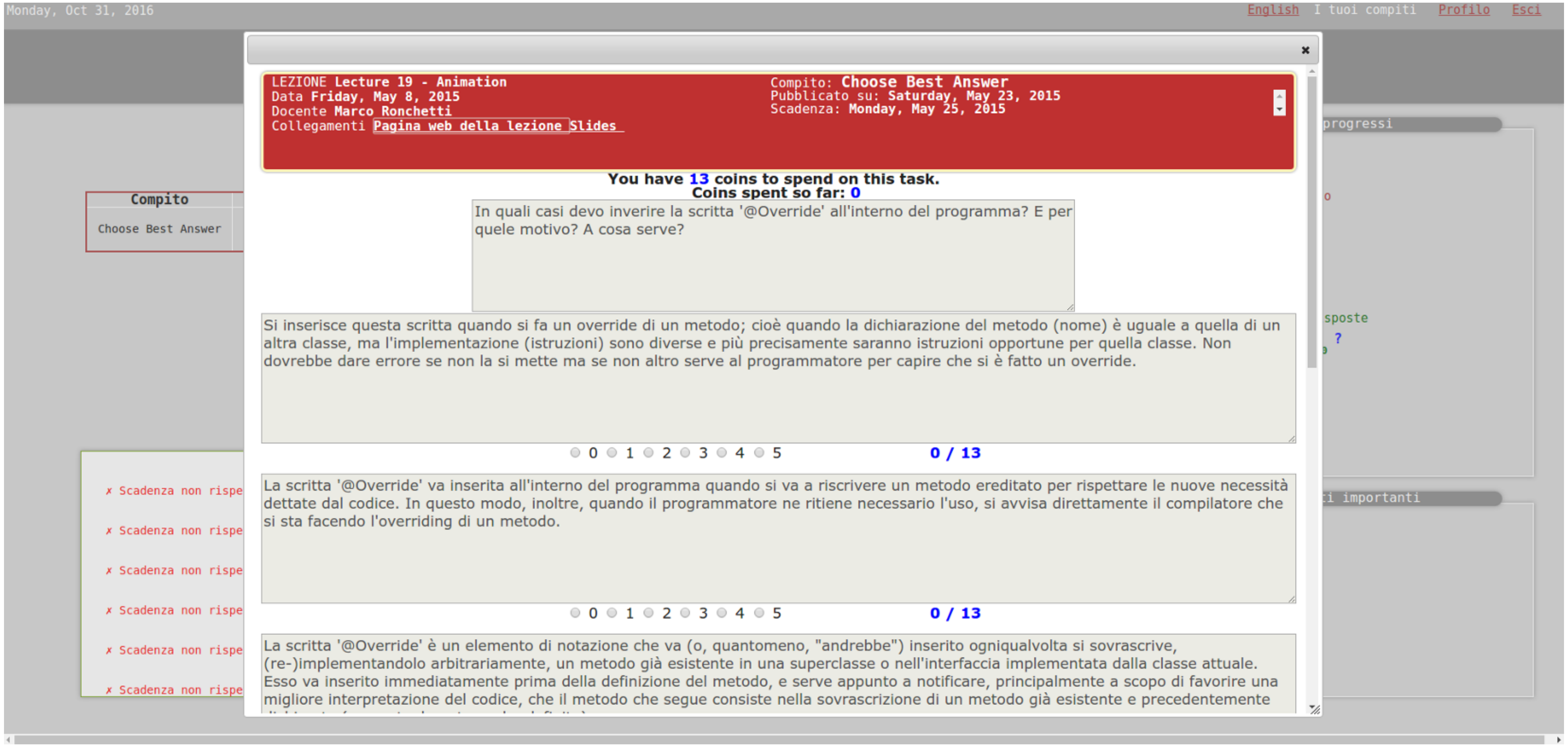
✗ Scadenza non rispettata per il compito Answer A Question, che era in Monday, May 11, 2015. Mostra

✗ Scadenza non rispettata per il compito Ask A Question, che era in Thursday, May 7, 2015. Mostra

✗ Scadenza non rispettata per il compito Choose Best Answer, che era in Monday, May 4, 2015. Mostra

Collegamenti importanti

Figure 3.11: Peer-Assessment System - Student Homepage 3



LEZIONE **Lecture 19 - Animation**
Data **Friday, May 8, 2015**
Docente **Marco Ronchetti**
Collegamenti **Pagina web della lezione** **Slides**

Compito: **Choose Best Answer**
Pubblicato su: **Saturday, May 23, 2015**
Scadenza: **Monday, May 25, 2015**

**You have 13 coins to spend on this task.**
**Coins spent so far: 0**

In quali casi devo inverire la scritta '@Override' all'interno del programma? E per quele motivo? A cosa serve?

Si inserisce questa scritta quando si fa un override di un metodo; cioè quando la dichiarazione del metodo (nome) è uguale a quella di un altra classe, ma l'implementazione (istruzioni) sono diverse e più precisamente saranno istruzioni opportune per quella classe. Non dovrebbe dare errore se non la si mette ma se non altro serve al programmatore per capire che si è fatto un override.

○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ 5 **0 / 13**

La scritta '@Override' va inserita all'interno del programma quando si va a riscrivere un metodo ereditato per rispettare le nuove necessità dettate dal codice. In questo modo, inoltre, quando il programmatore ne ritiene necessario l'uso, si avvisa direttamente il compilatore che si sta facendo l'overriding di un metodo.

○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ 5 **0 / 13**

La scritta '@Override' è un elemento di notazione che va (o, quantomeno, "andrebbe") inserito ogniqualvolta si sovrascrive, (re-)implementandolo arbitrariamente, un metodo già esistente in una superclasse o nell'interfaccia implementata dalla classe attuale. Esso va inserito immediatamente prima della definizione del metodo, e serve appunto a notificare, principalmente a scopo di favorire una migliore interpretazione del codice, che il metodo che segue consiste nella sovrascrizione di un metodo già esistente e precedentemente

Figure 3.12: Peer-Assessment System - Student Task Completion

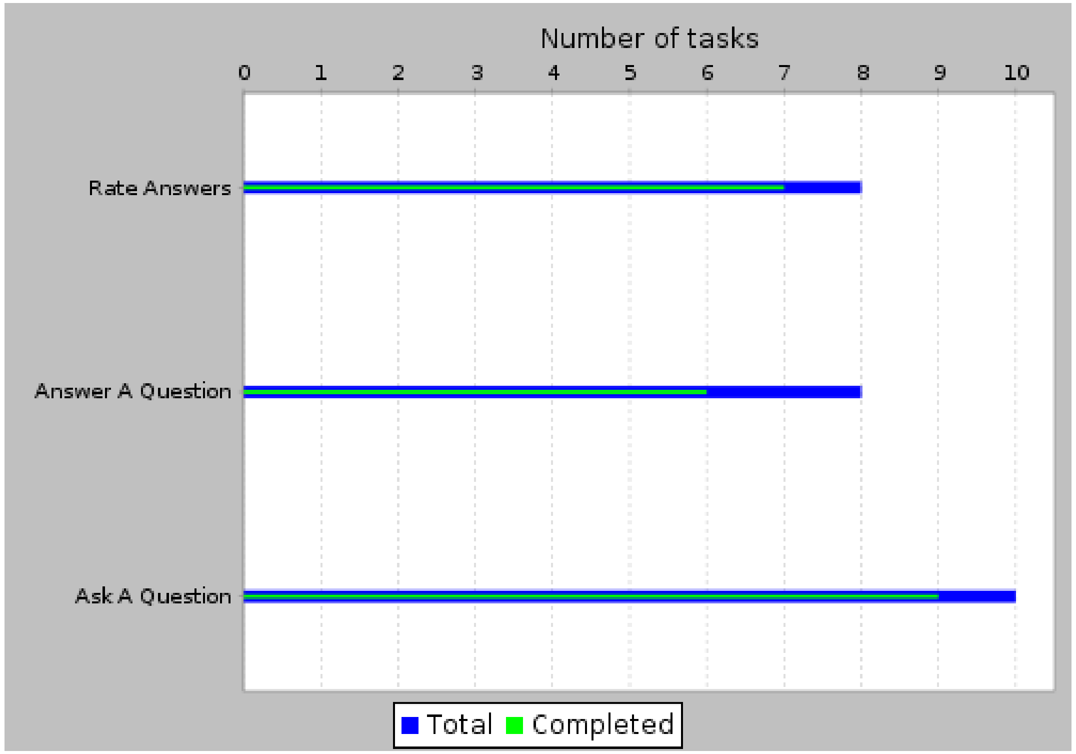
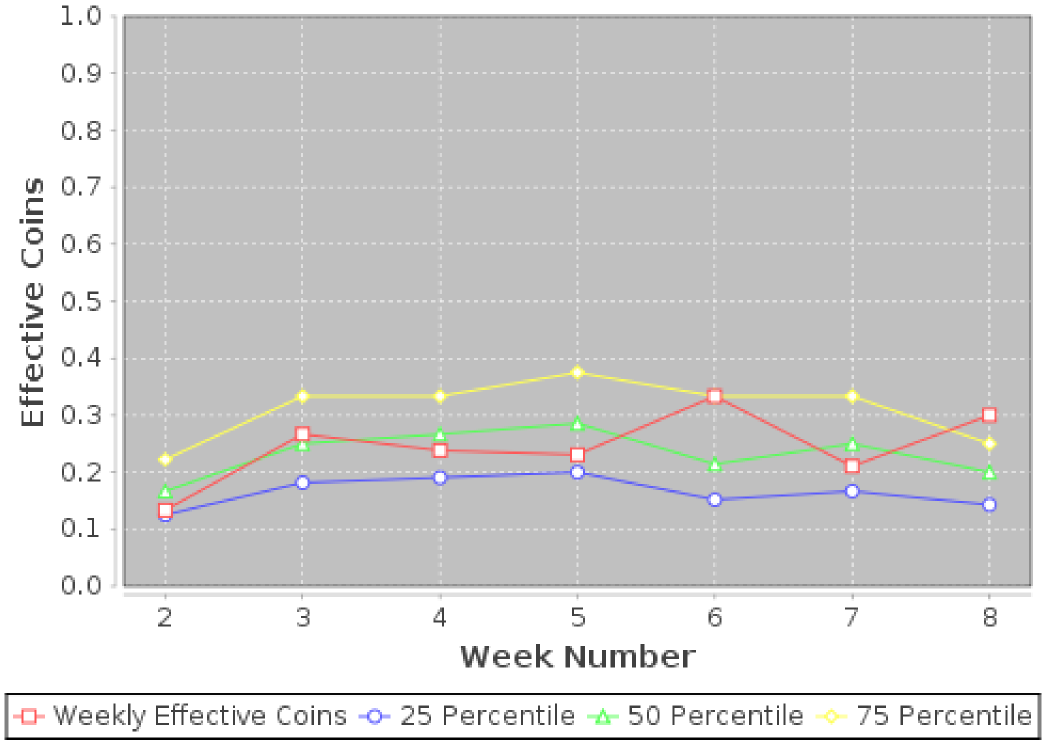


Figure 3.13: Peer-Assessment System - Student Weekly Performance

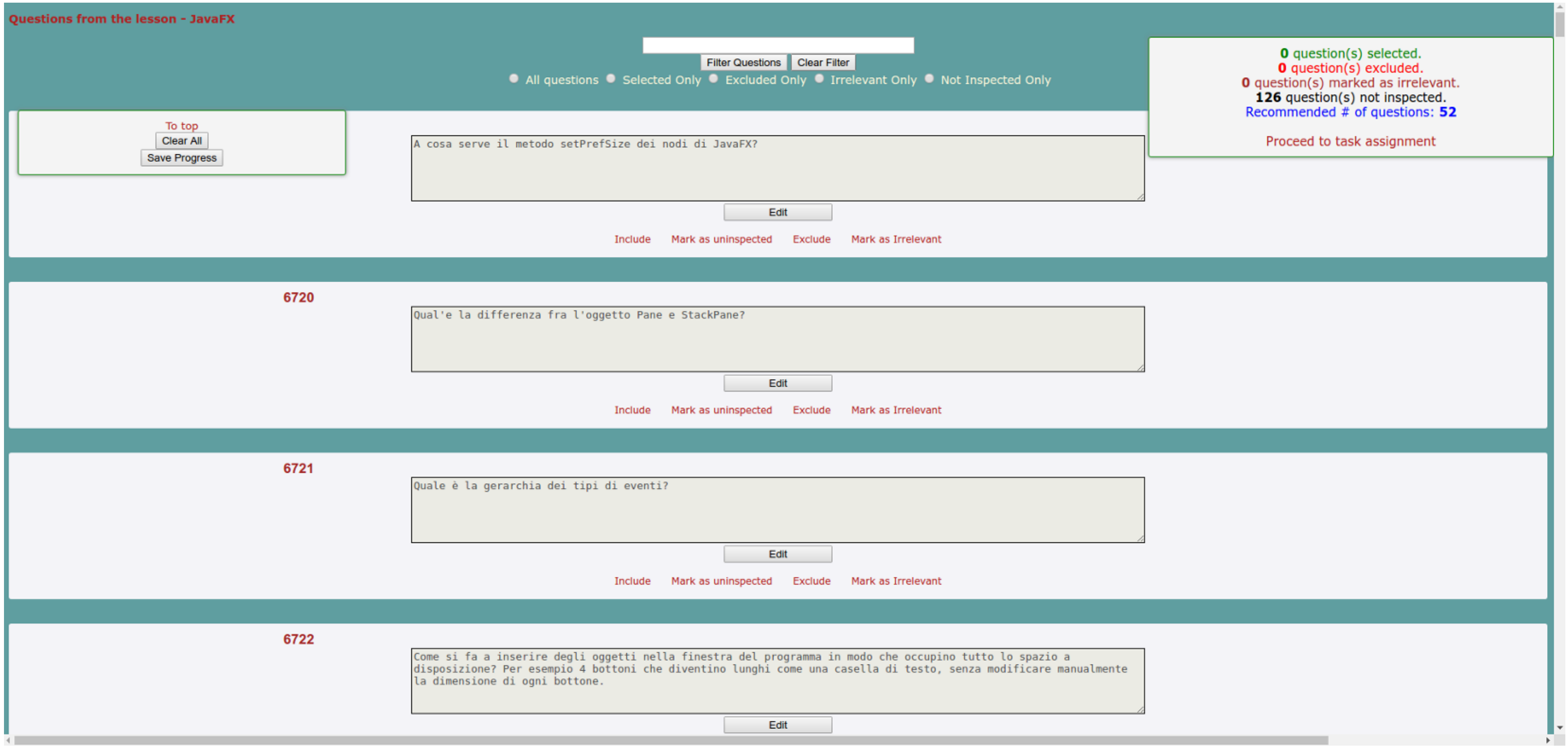


### 3.3.3 The Question Selection Module

This module was designed to assist the instructor in selecting a subset of questions collected during the 'Ask a Question' task phase to be distributed to students. The question selection process was semi-automated by implementing a question clustering feature using the K-Means clustering algorithm to group similar questions. The clustering algorithm used Term Frequency-Inverse Document Frequency (tf-idf) and Cosine Similarity to group questions.

Figure 3.14 provides a snapshot of this module.

Figure 3.14: Peer-Assessment System - Supervisor Question Selection Page


Questions from the lesson - JavaFX
Filter Questions
Clear Filter
○ All questions ○ Selected Only ○ Excluded Only ○ Irrelevant Only ○ Not Inspected Only
To top
Clear All
Save Progress
Edit
Include
Mark as uninspected
Exclude
Mark as Irrelevant
Proceed to task assignment

**0** question(s) selected.
**0** question(s) excluded.
**0** question(s) marked as irrelevant.
**126** question(s) not inspected.
Recommended # of questions: **52**

A cosa serve il metodo setPrefSize dei nodi di JavaFX?

**6720**

Qual'e la differenza fra l'oggetto Pane e StackPane?

**6721**

Quale è la gerarchia dei tipi di eventi?

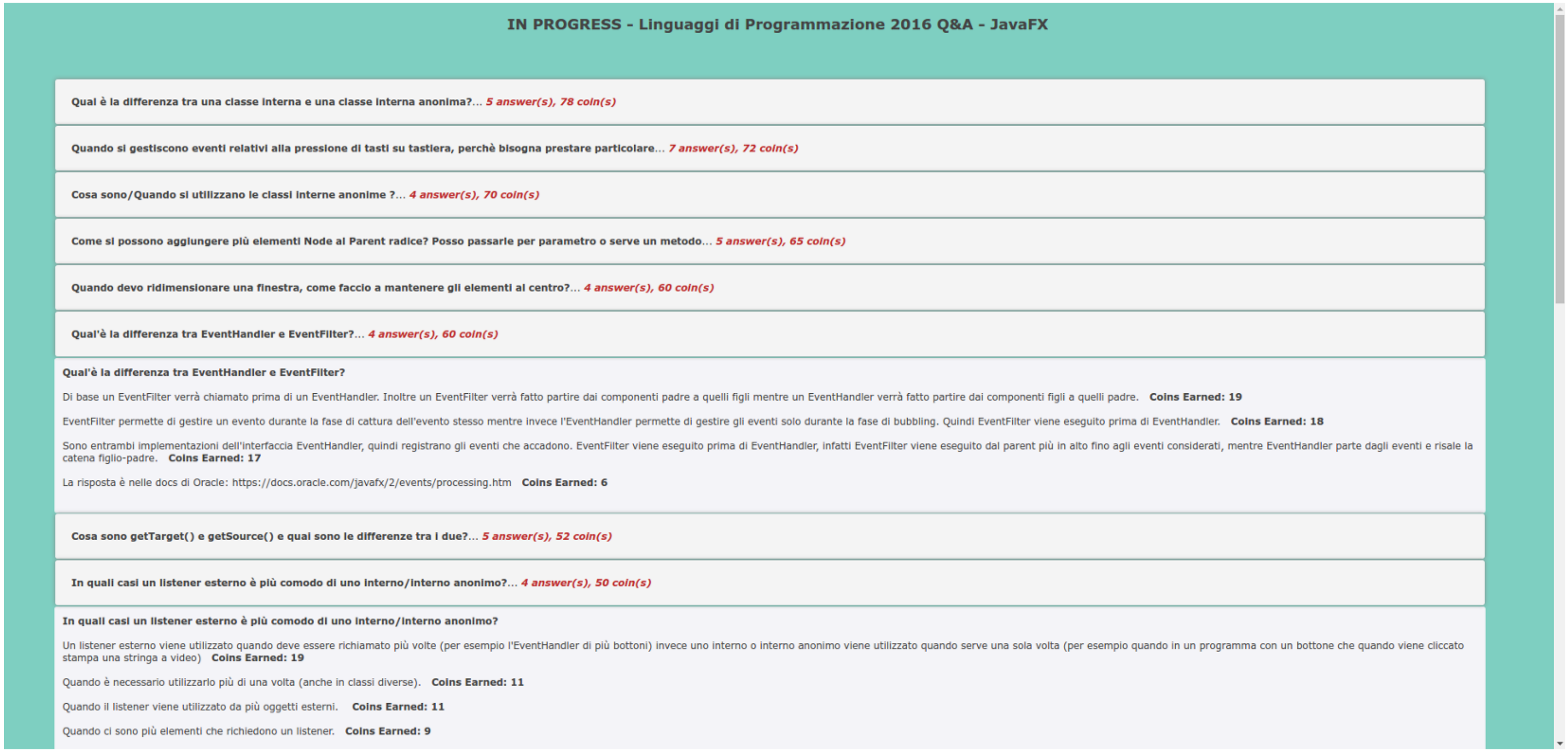
**6722**

Come si fa a inserire degli oggetti nella finestra del programma in modo che occupino tutto lo spazio a disposizione? Per esempio 4 bottoni che diventino lunghi come una casella di testo, senza modificare manualmente la dimensione di ogni bottone.

### 3.3.4 The Q&A Summary Page

This feature was designed to provide students with a weekly summary of the Q&A sets. The Q&A items are organized as collapsible tiles with the question, the number of answers and total points earned by all answers to the question reported in the header of each tile.

Figure 3.15 provides a snapshot of this module.



**IN PROGRESS - Linguaggi di Programmazione 2016 Q&A - JavaFX**

**Qual è la differenza tra una classe interna e una classe interna anonima?**... *5 answer(s), 78 coin(s)*

**Quando si gestiscono eventi relativi alla pressione di tasti su tastiera, perchè bisogna prestare particolare**... *7 answer(s), 72 coin(s)*

**Cosa sono/Quando si utilizzano le classi interne anonime ?**... *4 answer(s), 70 coin(s)*

**Come si possono aggiungere più elementi Node al Parent radice? Posso passarle per parametro o serve un metodo**... *5 answer(s), 65 coin(s)*

**Quando devo ridimensionare una finestra, come faccio a mantenere gli elementi al centro?**... *4 answer(s), 60 coin(s)*

**Qual'è la differenza tra EventHandler e EventFilter?**... *4 answer(s), 60 coin(s)*

**Qual'è la differenza tra EventHandler e EventFilter?**

Di base un EventFilter verrà chiamato prima di un EventHandler. Inoltre un EventFilter verrà fatto partire dai componenti padre a quelli figli mentre un EventHandler verrà fatto partire dai componenti figli a quelli padre. **Coins Earned: 19**

EventFilter permette di gestire un evento durante la fase di cattura dell'evento stesso mentre invece l'EventHandler permette di gestire gli eventi solo durante la fase di bubbling. Quindi EventFilter viene eseguito prima di EventHandler. **Coins Earned: 18**

Sono entrambi implementazioni dell'interfaccia EventHandler, quindi registrano gli eventi che accadono. EventFilter viene eseguito prima di EventHandler, infatti EventFilter viene eseguito dal parent più in alto fino agli eventi considerati, mentre EventHandler parte dagli eventi e risale la catena figlio-padre. **Coins Earned: 17**

La risposta è nelle docs di Oracle: https://docs.oracle.com/javafx/2/events/processing.htm **Coins Earned: 6**

**Cosa sono getTarget() e getSource() e quai sono le differenze tra i due?**... *5 answer(s), 52 coin(s)*

**In quali casi un listener esterno è più comodo di uno interno/interno anonimo?**... *4 answer(s), 50 coin(s)*

**In quali casi un listener esterno è più comodo di uno interno/interno anonimo?**

Un listener esterno viene utilizzato quando deve essere richiamato più volte (per esempio l'EventHandler di più bottoni) invece uno interno o interno anonimo viene utilizzato quando serve una sola volta (per esempio quando in un programma con un bottone che quando viene cliccato stampa una stringa a video) **Coins Earned: 19**

Quando è necessario utilizzarlo più di una volta (anche in classi diverse). **Coins Earned: 11**

Quando il listener viene utilizzato da più oggetti esterni. **Coins Earned: 11**

Quando ci sono più elementi che richiedono un listener. **Coins Earned: 9**

## 3.4 Reception of the Peer-Assessment Platform by Students

The peer-assessment framework was used by over 600 students enrolled in five courses between the academic years 2013/14 and 2015/16. The majority of these courses were offered at an undergraduate level and all were courses from the department of computer science. The first three courses in which the system was used were Informatica Generale I (IG1), a first year bachelor course, Programmazione Android (PA), a third year bachelor course, and Web Architectures (WA), a course for first year master students. A total of 382 students participated in online peer-assessment tasks for these courses.

The most recent version of the system was utilized in an Object Oriented Programming course offered to 150 first-year computer science students and about 31 third-year math students, for a total of 181 students.

In order to understand students perception of the peer-assessment approach and to explore whether it had elicited engagement and continuous revision of the course material, students were asked to complete questionnaires.

For each of the first three courses, the questionnaire included 13 five-point Likert items, 1 multiple-choice, multiple-answer question and 3 open-ended questions. More than half of the students responded for each course - 124 out of 222 (55%) for IG1, 80 out of 120 (66%) for PA, and 23 out of 40 (57%) for WA.

Considering the size of the statistical samples and under the assumptions of a simple random sampling of the population, a sample proportion of 50%, and a finite population, the respective margins of error at 95% confidence level were computed as 6% for IG1, 6% for PA and 13% for WA.

The initial set of questions sought to understand if the peer-assessment system was accepted by students and if it was perceived as an effective tool of learning. Students were asked if they thought asking questions and responding to their peers' questions was useful. The majority of students, between 45% ( ± 6% Margin of Error) and 87% ( ± 9% Margin of Error) responded positively for the three courses. An interesting observation here was that the percentage of positive responses increased significantly with the course level. It appeared that postgraduate students were very content with the approach.

Regarding engagement and attentiveness in class, first year undergraduate students of IG1 were closely divided on whether participation in online peer-assessment tasks enhanced their attentiveness in class (28% ± 5% for YES and 32% ± 6% for NO). This sentiment was reversed for third-year undergraduates - positive responses were relatively higher (40% ± 6%) than negative responses (31% ± 6%). Postgraduate students had much higher positive responses (65% ± 13%). Concretely, a positive correlation was witnessed between perceived increase in attentiveness brought about by participation in online peer-assessment tasks and course level.

The system was also deemed to be effective in providing a push for students to follow their course more regularly, with more positive responses reported at higher course levels. First year undergraduate students were divided at 39% ± 6% (positive) to 30% ± 5% (negative), while 83% ± 10% of postgraduates responded positively.

Similar trends were witnessed in students responses regarding whether involvement in peer-assessment tasks induced deeper studying and better preparation for the final exam. At least 43% ± 6% of first-year undergraduates thought it had a positive effect on their study and preparation for the exam, while at least 39% ± 6% of third-year undergraduates thought the same. At least 69% ± 12% of postgraduates had the same view.

Regarding the amount of work introduced by the peer-assessment framework, the vast majority of students thought it was adequate - only

14% ± 4%, 10% ± 4% and 13% ± 9% responding otherwise for IG1, PA and WA, respectively. Given their smaller number, a higher proportion of postgraduates thought the additional workload was burdensome. This may be explained by the higher workload that postgraduate students already have.

The findings of the evaluation results were corroborated by those from a recent evaluation of the system. The latest evaluation, from the academic year 2015/16 was thorough. Students enrolled in an Object Oriented Programming course using the peer-assessment framework were asked to complete an online survey of the system and of the overall approach before sitting the final exam. Of the 150 students who were enrolled at the beginning of the course, 117 completed the survey.

The first point of focus of this survey was exploring how students perceived being evaluated by their peers. This has been one of the most popular topics in peer-assessment research for decades. It was thought worthy to understand whether the student responses agreed with previous findings. Students were explicitly asked if they found the practice to be an injustice. Responses were similar to many inconclusive findings in the literature in that no strong opinion was expressed by a majority. 33% ± 5% agreed it was fair whereas 34 ± 5% believed it was unfair. The rest did not respond to this question.

A follow-up question found that only 5% ± 2% felt uneasy about the practice of being evaluated by their peers and 25% ± 4% did not like this form of assessment. 55% ± 5% said they found it useful and only 18% ± 4% thought it was not useful. Asked about the usefulness of such practice as a form of formative assessment, the large majority (60% ± 5%) agreed it was so and only 13% ± 3% thought otherwise.

The perceived appropriateness of the introduced workload by the framework was explored here as well, although in greater detail using diverse phrases and several questions. 61% ± 5% responded that the required extra effort was not excessive while only 11% ± 3% responded that it was excessive.11% ± 3% believed the time they dedicated to using the system was badly spent whereas 69% ± 5% believed it was worth it. Similarly, the time spent on using the system was deemed unproductive by 6% ± 3% of the respondents and productive by 63% ± 5% of the respondents. Using the system was perceived to be a nuisance for 15% ± 3% of the respondents and not at all by 51% ± 5% of the respondents.

The utility of the approach was confirmed in this round of evaluations as well. 74% ± 4% of respondents thought the experience was useful, against a mere 10% ± 3%, who were convinced otherwise. An analysis of the reasons provided revealed that 67% ± 4% felt motivated to review the material covered by the lectures whereas only 9% ± 3% did not feel the need to do so. 55% ± 5% thought this prepared them better for the exam, in contrast with 10% ± 3% who did not see such advantage. 60% ± 4% believed being involved in peer-assessment tasks helped them to keep up with the pace of the course whereas 10% ± 3% did not believe so. However, only 22% ± 4% thought participation in online peer-assessment tasks made them more attentive in class in order to prepare for subsequent question posing and answering tasks. 26% ± 4% did not believe so and 52% had no opinion on the matter. This observation may be well explained by the fact that students largely relied on the teacher's lecture slides than either their notes or other resources in order to help them complete question posing and answering tasks. Indeed, 75% ± 4% used the lecture slides. 36% ± 4% always or often reviewed their own notes.

The fact that the teacher's slides were the most preferred source of information reinforced the earlier indication that the system pushed students to review the material presented in class throughout all peer-assessment task phases. It was thus concluded that the repeated reviewing of the material discussed in class, either in part or in its entirety, served as the raison d'être of the methodology.

Lastly, students were asked if they wished the peer-assessment framework to be applied in other courses. 61% ± 5% replied positively, as opposed to 20% ± 3% who did not wish to use the system in the future.

## 3.5 Conclusion

A peer-assessment methodology implementing a slightly competitive social game was discussed in this chapter. The goal of the peer-assessment methodology was to improve student engagement in courses with a large number of students. The web-based peer-assessment framework implementing this methodology was utilized in several courses since the 2013-14 academic year.

The design of online peer-assessment tasks makes the system suitable to be applied in very large classes, including Massive Open Online Courses (MOOCs).

Earlier indications of the system's acceptability were strongly confirmed in a recent evaluation of its latest version. The majority of students expressed satisfaction with the system's capacity to encourage continued revision practices and to promote better preparation for final exams. The large majority of students also believed in the utility of the online peer-assessment system as a formative approach to assessment and learning.

Multiple evaluations of the framework have supported the validity of the methodology behind it. Apart from a foreseen re-implementation of certain modules in order to address minor technical issues, it is believed that the system is ready to be widely deployed across departments at the University of Trento. Future evaluations of the system will also take into account the potential impact of certain factors on the methodology such as students' departments and course levels as well as basic demographic data such as gender and age groups.

# 4

# Online Peer-Assessment as a Tool of Early Intervention

## 4.1 Introduction

This chapter explores the applicability of online peer-assessment as a mechanism early intervention. It discusses whether the peer-assessment methodology and web-based framework discussed in the previous chapter could yet play another role in early identification of students that risk failing courses.

In order to determine if this was the case, the relationship between online activity and performance in final exams was studied. The study examined online peer-assessment data gathered from three undergraduate-level computer science courses offered between early 2013 and mid 2016. Analysis of the peer-assessment data and final exam scores of a total of 619 students for the three courses revealed that there was a correlation between low task completion rate and course incompletion. However, the analysis did not find any strong relationship between high task completion rate and successful course completion.

## 4.2 Description of the Peer-Assessment Data

Although participation in peer-assessment tasks was optional, all students who completed at least a third of the tasks were awarded a bonus worth 3.3% of the final mark. An additional 3.3% bonus was awarded to the top-third students, based on the number of peer-awarded points. For all three courses, it was observed that active participation in peer-assessment activities declined towards the end of the course. Regardless, a total of 83% of students for the three courses completed at least a third of the tasks.

How to predict expected student performance using student activity data from the peer-assessment system is explored in the two chapters that proceed. The prediction models discussed there, however, considered only those students who had completed over a third of all peer-assessment tasks. The main reason behind this was that the performance of the predictive models was significantly reduced with the introduction of data of students with little or no participation at all. Because the number of students who did not participate enough in online peer-assessment tasks was considerably low, the attempt to build prediction models only for those students did not produce encouraging results.

Therefore, the analyses presented here used less sophisticated statistics to perform comparisons between the two student groups. Although it may seem appropriate to defer this discussion until the predictive models have been presented, it was placed here with the belief that the approach followed in this chapter has more in common with that presented in the previous chapter.

The three undergraduate-level computer science courses were labelled IG1, LP, and PR2. The Italian grading system uses a scale that ranges between 0 and 30, with 30L or 30 Excellent the highest possible mark. In order to pass a course, students have to obtain at least a score of 18. For the purpose of this analysis, the range of scores was categorized into

Table 4.1: Mapping of Scores to Performance Groups

| **Score** | **Verdict** |
|---|---|
| Below 18 | Insufficient |
| Between 18 and 22 | Low performer |
| Between 23 and 26 | Medium performer |
| 27 or above | High performer |

Table 4.2: Distribution of Scores for Students in the Low-Participation Group

| **Course** | **Number of Students** | **<18** | **[18.23)** | **[23, 26]** | **>=27** |
|---|---|---|---|---|---|
| IG1 | 35 | 25 | 4 | 3 | 3 |
| PR2 | 42 | 18 | 7 | 11 | 6 |
| LP | 30 | 22 | 1 | 5 | 2 |

Table 4.3: Distribution of Scores for Students in the High-Participation Group

| **Course** | **Number of Students** | **<18** | **[18.23)** | **[23, 26]** | **>=27** |
|---|---|---|---|---|---|
| IG1 | 182 | 69 | 46 | 36 | 31 |
| PR2 | 141 | 23 | 33 | 49 | 36 |
| LP | 189 | 84 | 40 | 37 | 28 |

four groups and labels were assigned to each group. Table 4.1 presents this partitioning of scores. Students had the opportunity to improve their grades by making several attempts. The analysis considers the data of those students who both subscribed for peer-assessment tasks and sat the exam at least once.

## 4.3 Analysis and Results

For each course, students were divided into low-participation and high-participation groups on the 33% task completion mark, with the former falling below that mark. A further categorization was then made to explore the intersection between each participation group and exam performance group. Tables 4.2 and 4.3 present the number of students in each intersection.

It was observed that, across all three courses, a large majority of low-participation students did not manage to obtain passing marks. Although the course incompletion rate of low-participation students enrolled in the course PR2 was not the highest across all three courses, low participation students here were more than twice as likely to score below the passing mark as their high participation counterparts. Similarly, LP and IG1 low-participation groups were 1.66 and 1.87 times as likely as their high-participation counterparts to score below the passing mark, respectively.

Another observation that emerged from analysis of the data was that low-participation students usually became inactive within the first four weeks, before halfway through the courses. This led to much of the data for low-participation students changing very little throughout the remainder of the courses. This important observation strengthened the argument that the online peer-assessment system could be used as a tool for identifying potentially at-risk students as early as four weeks into the courses.

Yet another observation was the large difference in the percentage of students with insufficient performance between the low-participation and high-participation groups. This difference ranged between 27% and 33% for the three courses. Although with varying degrees, this confirmed that low-participation students had a much lesser chance of successfully completing the courses. The following charts demonstrate in greater detail, the differences in performance levels between low-participation and high-participation groups for the three courses.

Figure 4.1: Participation in Peer-Assessment Tasks and Exam Scores for Course IG1

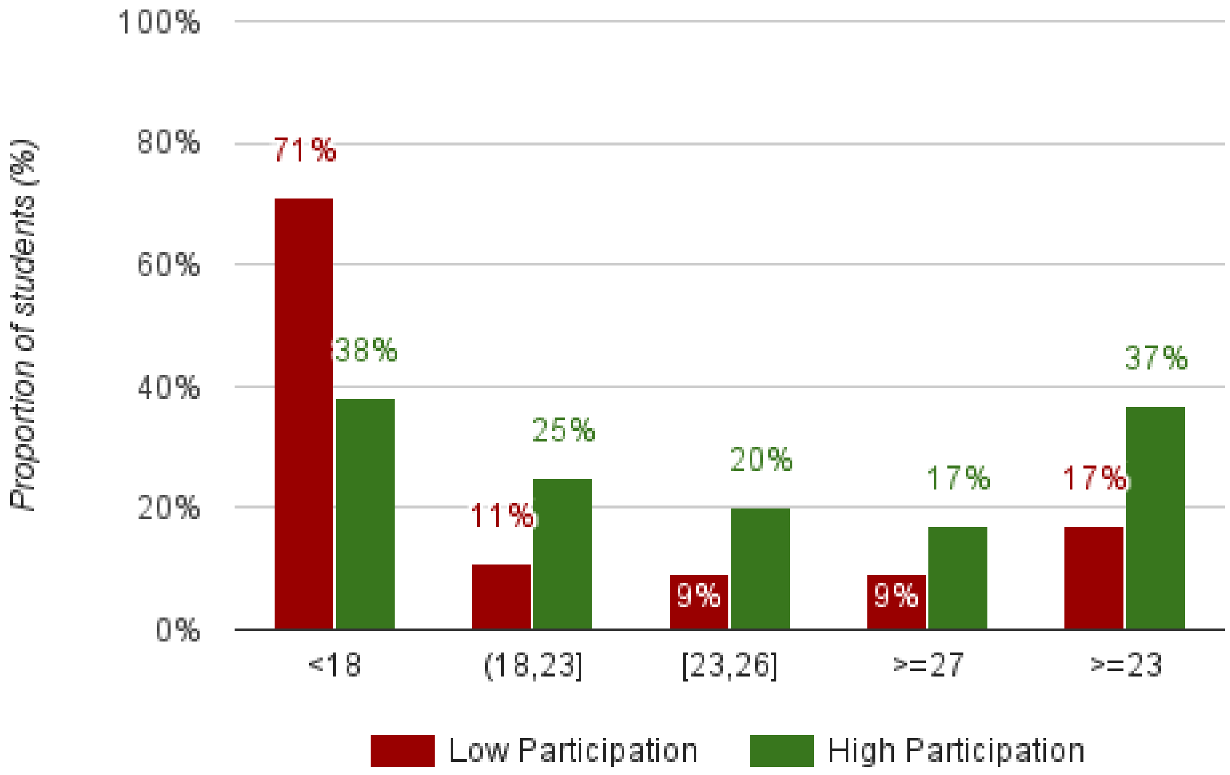


Figure 4.2: Participation in Peer-Assessment Tasks and Exam Scores for Course PR2

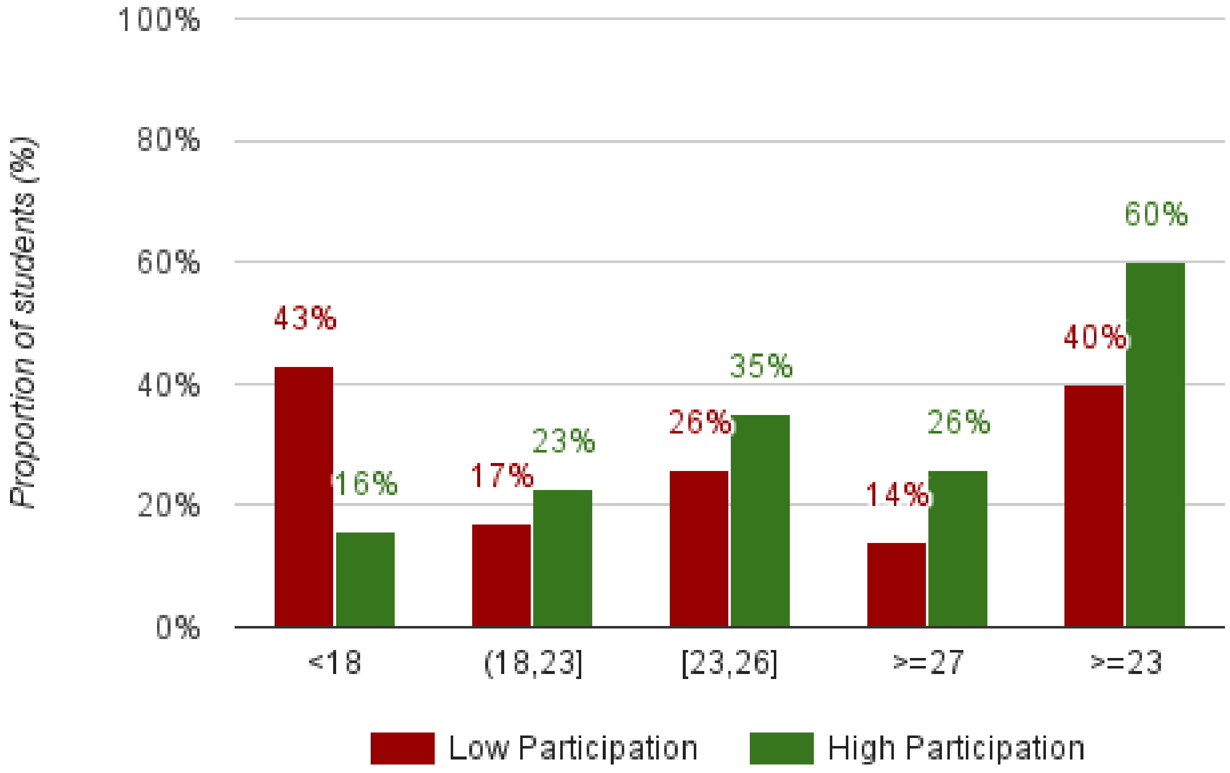
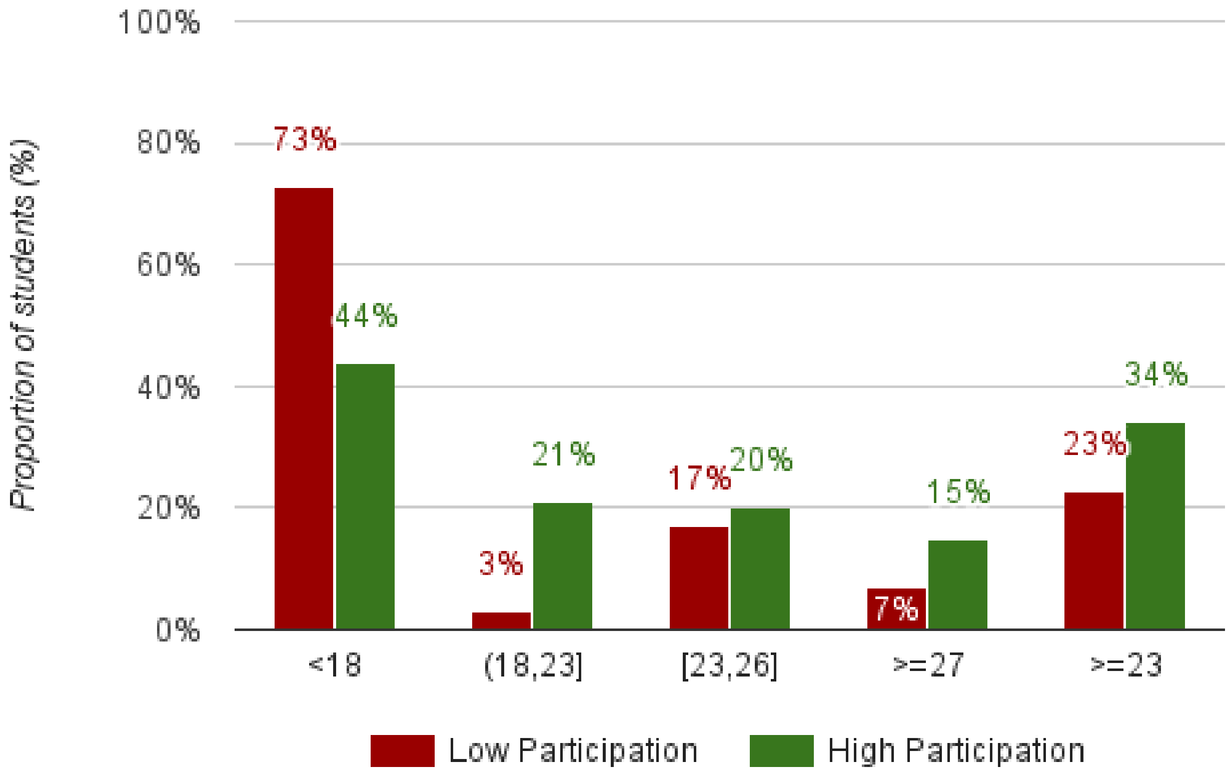


Figure 4.3: Participation in Peer-Assessment Tasks and Exam Scores for Course LP



## 4.4 Conclusion

It was sought to determine if the peer-assessment methodology and the web-based framework discussed in the previous chapter could yet play a role other than promoting student engagement.

Data from over 600 students was analyzed to determine if there was a correlation between low participation in online peer-assessment tasks and course incompletion. The analysis revealed that the majority of students with little participation in online peer-assessment tasks struggled to either pass their exams or perform well. The findings contribute to yet another motivation for automated peer-assessment. Although, at this stage, there is not enough evidence to suggest that participation in online peer-assessment tasks improves overall student performance, the argument that these activities could provide well-timed identification of potentially at-risk students is not far-fetched.

A future extension of this work will focus on whether further categorization and mapping of students into more performance groups could provide better insights with respect to identifying students that may not be at risk but may still need closer supervision.

The case for introducing electronic peer-assessment environments into the classroom is supported by the foreseen significant improvements in efficiency and effectiveness of the activities involved. It is hoped that the prospects explored in this study contribute to the case for transitioning into cost-effective, ubiquitous, and highly interactive electronic peer-assessment solutions.

In particular, the transition to a ubiquitous system can be made with little difficulty by taking advantage of the fact that virtually all students own smartphones or tablets. Developing a mobile peer-assessment solution has the potential to increase student productivity given that peer-assessment tasks are designed to be simple, mobile-friendly and with special attention to privacy and other social aspects. Hence, development of a mobile version of the peer-assessment system will be addressed in the near future.

# 5

# Predicting Student Success from Peer-Assessment Activities

## 5.1 Introduction

A common approach to the problem of predicting students' exam scores has been to base this prediction on the previous educational history of students. As presented in the review of the literature on student performance prediction, there are no prediction techniques that utilize data from a peer-assessment environment, automated or otherwise, in order to predict performance on final exams. This chapter goes further and applies linear regression to peer-assessment data collected using the online system to determine if performance in peer-assessment activities can fairly determine expected student performance in final exams. By doing so, it aims to strengthen the argument that the peer-assessment methodology can be applied to classes with a large number of students, where close supervision of each student is hardly possible.

In order to build the prediction model, peer-assessment data from two undergraduate-level computer science courses, labelled IG1 and PR2, was used.

If carried out in a timely manner, automated prediction of final exam scores could provide information crucial for early identification of potentially at-risk students. Automated score prediction could also have a significant implication in the Massive Open Online Courses (MOOC) arena. Predicting student performance could help provide early insight into the attrition rates of courses administered in MOOC format.

Here, how student performance can be predicted by extracting student activity data is explored in detail. Although the datasets used in this study are not enthusiastically large, the fact that they have been collected from two separate courses, with each having at least a hundred students, has led to the argument that online peer-assessment can indeed serve as an alternative and efficient data source for performance prediction models.

The aim of this chapter is thus to promote the argument that data collected from the peer-assessment system could be used to build student performance prediction models, hence strengthening the role of the methodology, and peer-assessment in general, as tools of early intervention.

## 5.2 Building the Prediction Models

The peer-assessment data discussed in this chapter were collected using an earlier version of the web-based framework which included three main tasks: asking questions, answering questions and voting for the best answer among those provided by peers to a question. The data came from two courses. The courses were Informatica Generale I (IG1) and Programmazione 2 (PR2), introductory-level computer science courses offered to undergraduate students at the university of Trento. The three tasks were repeated every week, from the second to the final week of the courses. Of all the questions that were submitted during the 'Ask a Question' phase for each week, the teacher selected a subset that was used in the remaining tasks. When providing an answer to a question, students were also given the option to rate the difficulty, relevance and interestingness of the questions on a 1 to 5 Likert-scale.

All student activity including time spent on completing tasks was logged by the system. The university's exam scoring system uses a 0-30 scale where 30 excellent is the highest achievable score, with the minimum passing mark set to 18. The data used here come only from those students who passed the course. As explained in the previous chapter, the reason for this was that inclusion of data from students who had little or no participation in online peer-assessment tasks or who had below passing marks made the prediction models perform poorly. Therefore, the prediction models discussed here use student activity in the online peer-assessment tasks to predict the final scores of students on a scale of 18 to 30. Consequently, although a total of over 400 students participated in the online peer-assessment activities for the two courses, only the data of a total of 206 students were used to build the prediction models.

Intuitively, it was first sought to determine if the number of a student answers that were chosen as the best by peers could indicate how a student would perform in the final exam.

Hence, a preliminary investigation was conducted to determine if the number of answers chosen as the best, referred to as votes earned, could be used as a sole predictor of final scores. It was learned that the data did not indicate any such relationship. It was therefore decided to proceed with investigation of the impact that other variables obtained from the peer-assessment data may have on final scores.

This investigation sought to determine if the number of tasks a student completed throughout the course, the number of questions they submitted and answered and the difficulty of the questions they answered could be used as performance indicators.

Therefore, an initial list of seven parameters was used to build the linear regression models discussed here. An additional 16 parameters, most of which were aggregates of the initial seven variables, were also considered, among which an additional 7 were chosen.

The final 14 variables, referred to as features from hereon, that were used to build the prediction models are:

- **Tasks Assigned (TA)** – The number of tasks that were assigned to the student
- **Tasks Completed (TC)** – The number of tasks that the student completed
- **Questions Asked (QAS)** – The number of 'Ask a Question' tasks the student completed
- **Questions Answered (QAN)** – The number of 'Answer a Question' tasks the student completed
- **Votes Cast (VC)** – The number of 'Rate Answers' tasks the student completed
- **Questions picked for answering (QP)** – The number of the student's questions that were selected by the teacher to be used in 'Answer A Question' tasks
- **Votes Earned (VE)** – The number of votes the student earned for their answers
- **Votes Earned Total Difficulty (VED)** – The sum of the products of the votes earned for an answer and the difficulty level of the question, as rated by students themselves, for all answers submitted by the student
- **Votes Earned Total Relevance (VER)** – The sum of the products of the votes earned for an answer and the relevance level of the question, as rated by students themselves, for all answers submitted by the student
- **Votes Earned Total Interestingness (VEI)** – The sum of the products of the votes earned for an answer and the interestingness level of the question, as rated by students themselves, for all answers submitted by the student

- **Selected Q total difficulty (SQD)** – The sum of the difficulty levels of the student's questions, as rated by students themselves, which were selected to be used in subsequent tasks
- **Selected Q total relevance (SQR)** – The sum of the relevance levels of the student's questions, as rated by students themselves, which were selected to be used in subsequent tasks
- **Selected Q total interestingness (SQI)** – The sum of the interestingness levels of the student's questions, as rated by students themselves, which were selected to be used in subsequent tasks
- **Number of Attempts (NA)** – The number of attempts the student made to pass the course

The data were normalised using the min-max normalization method to convert each value into a value between 0 and 1. To perform this normalization and to build the linear regression models, the Weka data mining toolkit ([39]) was used. Three linear regression models were built. The first two were built per course and the third was built using the combined dataset for both courses. Although combining datasets from different courses may sound strange, this was done in order to explore if there were parameters that had similar effects across courses. The combination of the two datasets also allowed building a model with a much larger number of training sets.

The performance of each model was tested via 10-fold cross-validation. The Root Mean Squared Error (RMSE) was used as a performance evaluation metric, a common practice for evaluating the performance of linear regression models.

In order to investigate the impact of additional variables on the prediction of final scores, those variables were added to the initial model incrementally.

First, models were built using the initial 7 features. Then, more complex models were built by adding additional features step by step, and preserving the feature whenever its introduction increased performance. As a result, 3 sets of 7 linear regression models each were built.

Because discussing 13 different linear regression models, whose final performance was inferior, is lengthy and may deviate from the topic at hand, only the final model with the least RMSE is discussed here. This model was built using the 14 parameters discussed earlier and it is given by:

$$\mathrm{FS}(i) = \mathrm{C}^{\mathrm{T}}\mathrm{S}_i + 27.9$$

where S is a 14-by-n matrix (14 parameters by n students), $\mathrm{S}_i$ is the $i^{th}$ column in S representing student $i$, $\mathrm{C}^{\mathrm{T}}$ is the transpose of the column vector C given by:

$$C = \begin{bmatrix} -3.98 \\ -0.32 \\ \mathbf{0.63} \\ \mathbf{0.68} \\ -2.16 \\ \mathbf{0.10} \\ \mathbf{0.71} \\ \mathbf{22.92} \\ -16.29 \\ -5.02 \\ \mathbf{4.54} \\ -3.71 \\ 0 \\ -4.42 \end{bmatrix} \; and\, S_{\mathrm{i}} = \begin{bmatrix} TA_{\mathrm{i}} \\ TC_{\mathrm{i}} \\ \mathbf{QAS_i} \\ \mathbf{QAN_i} \\ VC_{\mathrm{i}} \\ QP_{\mathrm{i}} \\ \mathbf{VE_i} \\ \mathbf{VED_i} \\ VER_{\mathrm{i}} \\ VEI_{\mathrm{i}} \\ \mathbf{SQD_i} \\ SQR_{\mathrm{i}} \\ SQI_{\mathrm{i}} \\ NA_{\mathrm{i}} \end{bmatrix}$$

The features designated in bold in the coefficient and feature matrices indicate the parameters that had higher contributions than others. In addition to the votes earned, it was discovered that qualities such as the number of questions asked and answered and the difficulty levels of student questions selected by the teacher were strong indicators of final scores. It was also observed that having an answer for a difficult question chosen as the best was the strongest indicator of final scores. This supported the common hypothesis that students who have the capacity to provide the best answers to several difficult questions have a tendency to have high final scores. In the experiments discussed here, this tended to be true despite the fact that answers were rated by peers. This tendency also contributed positively to arguments in the peer-assessment literature in favor of the validity of peer-assigned grades.

Although an assertion on whether this is true across courses with differing content and number of students may not be made at this stage, the observations made here supported the common belief that students who perform well in homework and other online activities are likely to successfully complete the course, as shown in the previous chapter.

However, before concluding that the model was strong enough to predict final marks, its performance needed to be tested in a number of ways.

## 5.3 Evaluation and Results

The final model was built using only the data collected from IG1 as it had the lowest prediction errors, with a cross-validated RMSE of 2.93. This created the opportunity to test how the model would perform on data coming from another course. The model was thus tested with 101 instances from the PR2 course. The RMSE was found to be 3.44. This finding was very encouraging as the model could still perform very well when predicting final scores of students from another course. Although the two courses were different in that they were attended by different student groups, they still focused on two introductory-level programming languages. Thus, the performance of the model on data from a similar course provided a positive answer to the question whether one model could fairly predict performance in another course.

It should be noted that the goal behind constructing such a prediction model was not to determine whether a student would earn a specific score. Rather, the aim behind designing such models was to provide indications as to whether students were likely to be at-risk of failing, to have average performance or to perform very well. With respect to this aim, it is believed that the prediction model was indeed a fair indicator of the likelihood of several levels of success.

Another test sought to answer whether the model's predictions were easy to make. It was sought to explore whether these predictions were any better than random guessing. Hence, several mechanisms of random guessing were developed.

In order to simulate true random guesses that an actual person would make, each of these random guessing mechanisms was run 10000 times and the model's performance was compared with the average of these guesses.

First, a grade was assigned to each of the 206 students by randomly selecting a number from the valid range of grades, 18 to 30. This random assignment was performed 10000 times for students of both courses. The average RMSE of these assignments was then evaluated. The average RMSE for this technique was computed as 5.04.

Then, in order to simulate the guessing pattern of a human being with prior information about what the final scores of the courses would look like - that is, a human being that could make an educated guess - grades were sampled from previous editions of the courses. In order to conduct this sampling, several probability distributions that resembled the distributions of the grades for each course were examined. It was found that kernels were the best-fit probability distribution for each of the courses. Hence, samples were taken from these distributions 10000 times and RMSEs of all score assignments were averaged. Average RMSEs of 4.94 and 5.01 were obtained for IG1 and PR2, respectively.

Figures 5.1 and 5.2 show the distributions of actual scores from previous editions of the course, plotted against the probability density functions that resemble them best.

Figure 5.1: Histogram of the final scores of students of IG1 plotted against a kernel distribution

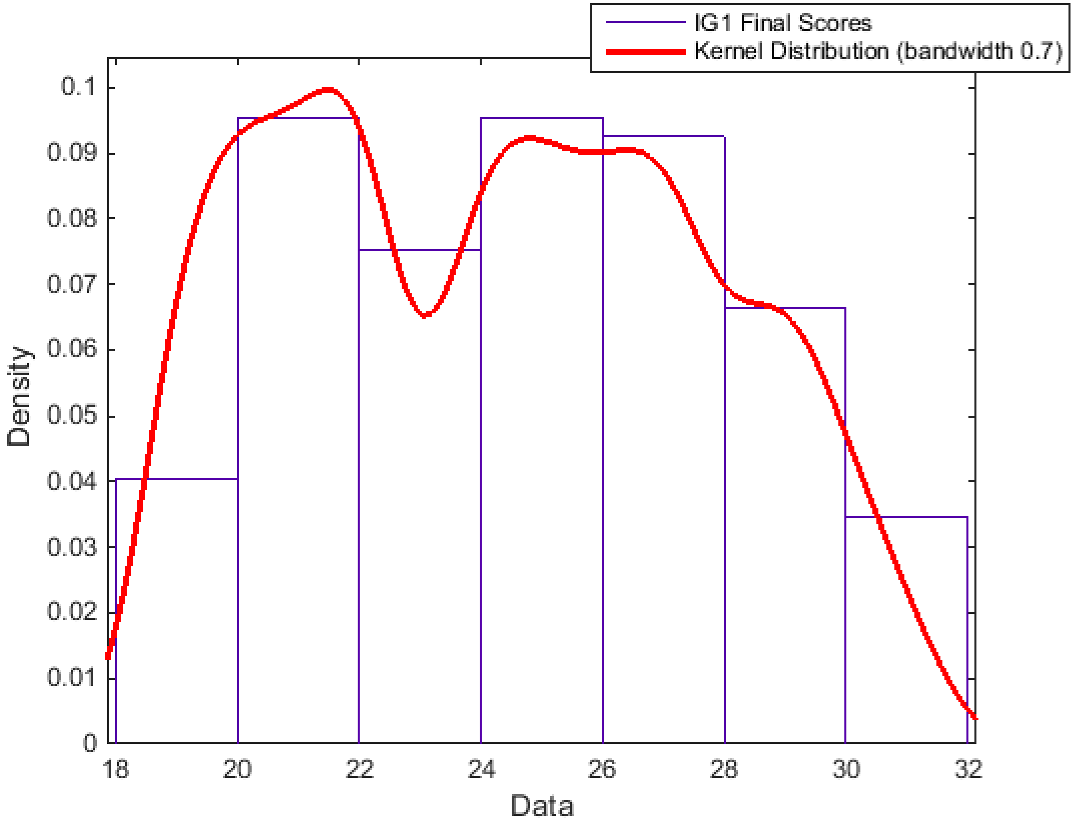


Figure 5.2: Histogram of the final scores of students of PR2 plotted against a kernel distribution

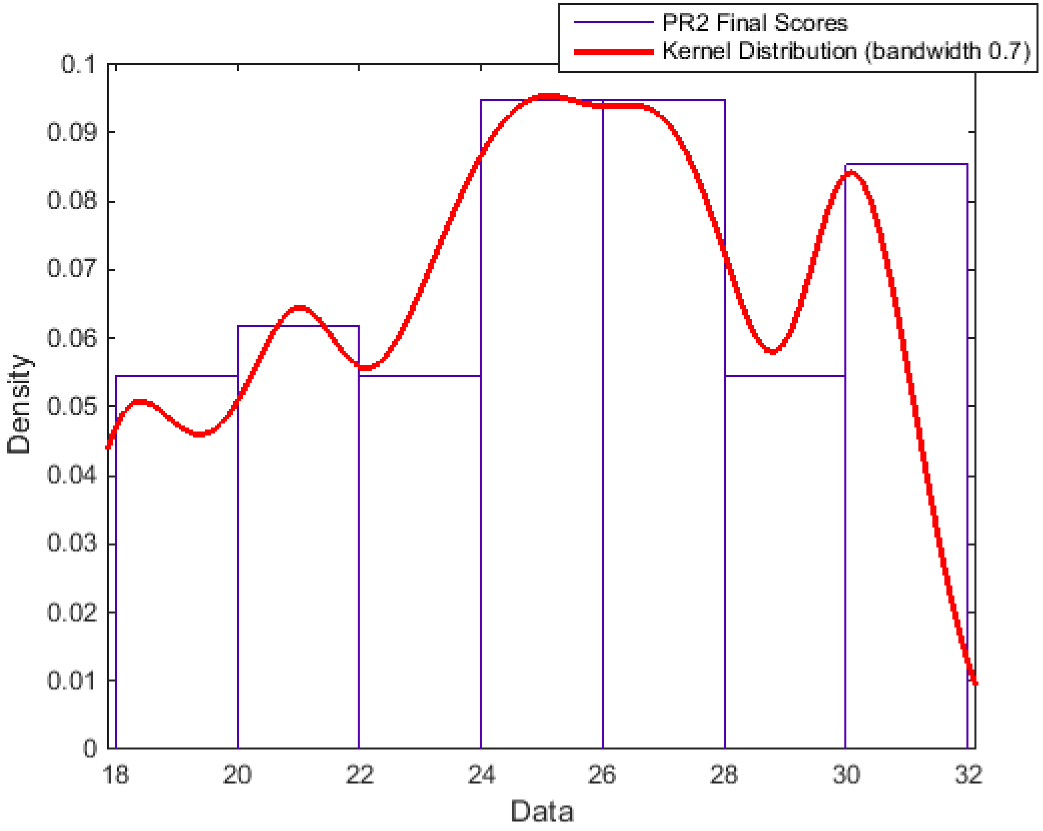


Table 5.1: Evaluation of prediction methods for IG1

| Prediction Method | RMSE |
|---|---|
| Sampling from a uniform distribution | 5.04 |
| Sampling from a kernel distribution | 4.94 |
| Sampling from previous scores directly | 4.89 |
| **Linear regression model** | **2.93** |

Table 5.2: Evaluation of prediction methods for PR2

| Prediction Method | RMSE |
|---|---|
| Sampling from a uniform distribution | 5.04 |
| Sampling from a kernel distribution | 5.01 |
| Sampling from previous scores directly | 4.96 |
| **Linear regression model** | **3.44** |

In the third round of tests, the sampling utilized a prior distribution that would very much resemble guessing by throwing a virtual dice with sides having one of each of N scores that were obtained from the previous editions of the course. In other words, random scores were assigned by rolling an N-sided dice, each of whose sides had a score written on it. Depending on the frequency of the scores in the actual distribution, the dice could have multiple sides with the same score written on them. For each of the 206 students, the dice was rolled 10000 times, resulting in 10000 sets of scores for 206 students. Each of these 10000 sets was then compared with actual scores to compute the RMSE. The average RMSE was then computed for the 10000 sets. This was computed as 4.89 for IG1 and 4.96 for PR2.

None of these rigorous tests resulted in predictions that were better or close to those made by the model. Indeed, the model's errors were significantly lower than those generated by any of the random guessing techniques, for both courses. Hence, it was concluded that the model's predictions were better than several techniques of random guessing.

Summaries of the evaluation results for both courses are presented in tables 5.1 and 5.2.

The histograms presented in figures 5.3 and 5.4 provide visual information about the distribution of the models errors. An ideal prediction model would have an error rate very close to zero. Hence, its histogram would have a very slender shape, with the peak located close to the center of the X-axis. As shown in the histograms, the errors of the model resemble such a distribution, with very low levels of error recorded as one moves away from the center of the X-axis.

Figure 5.3: Histogram of the prediction errors for IG1

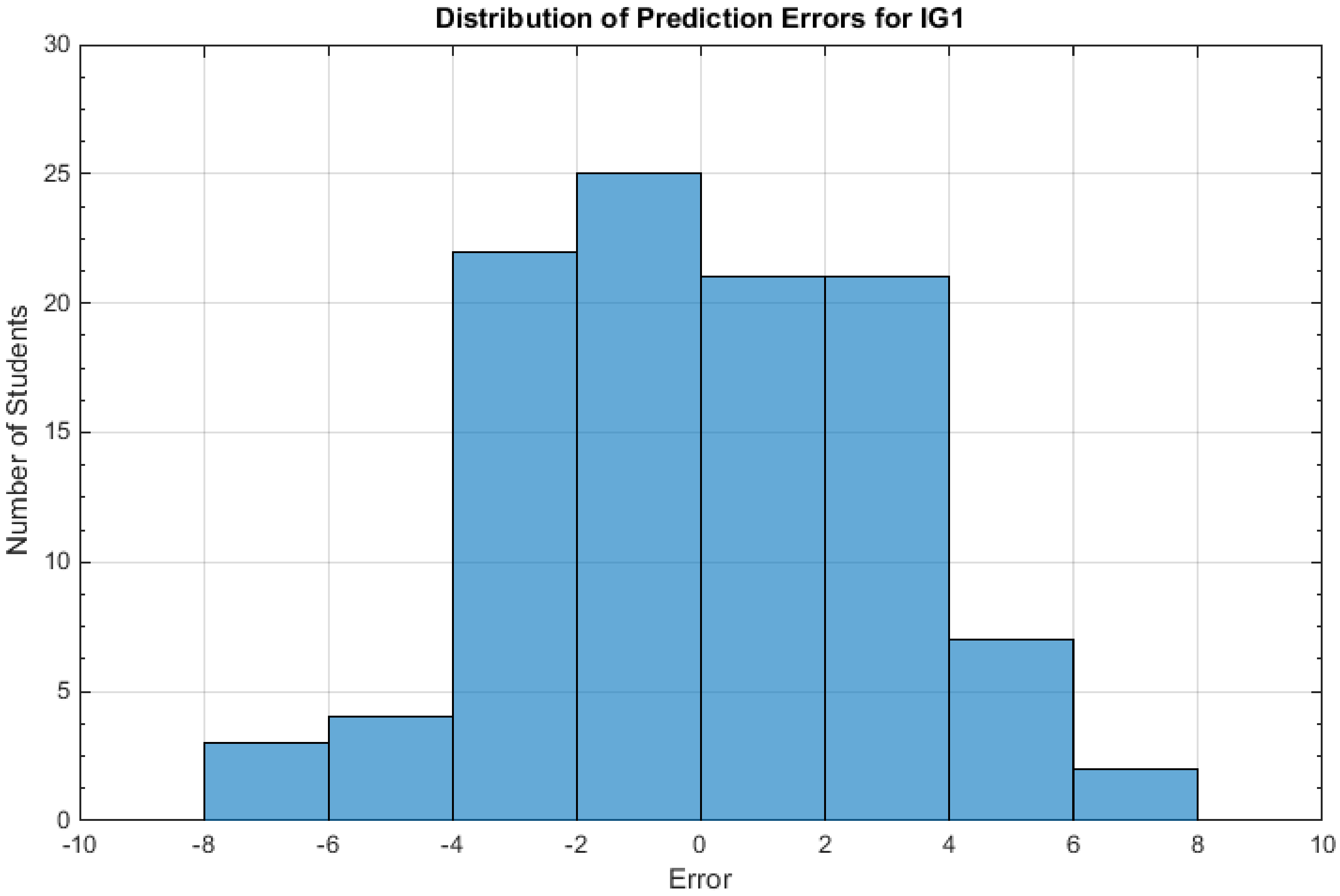
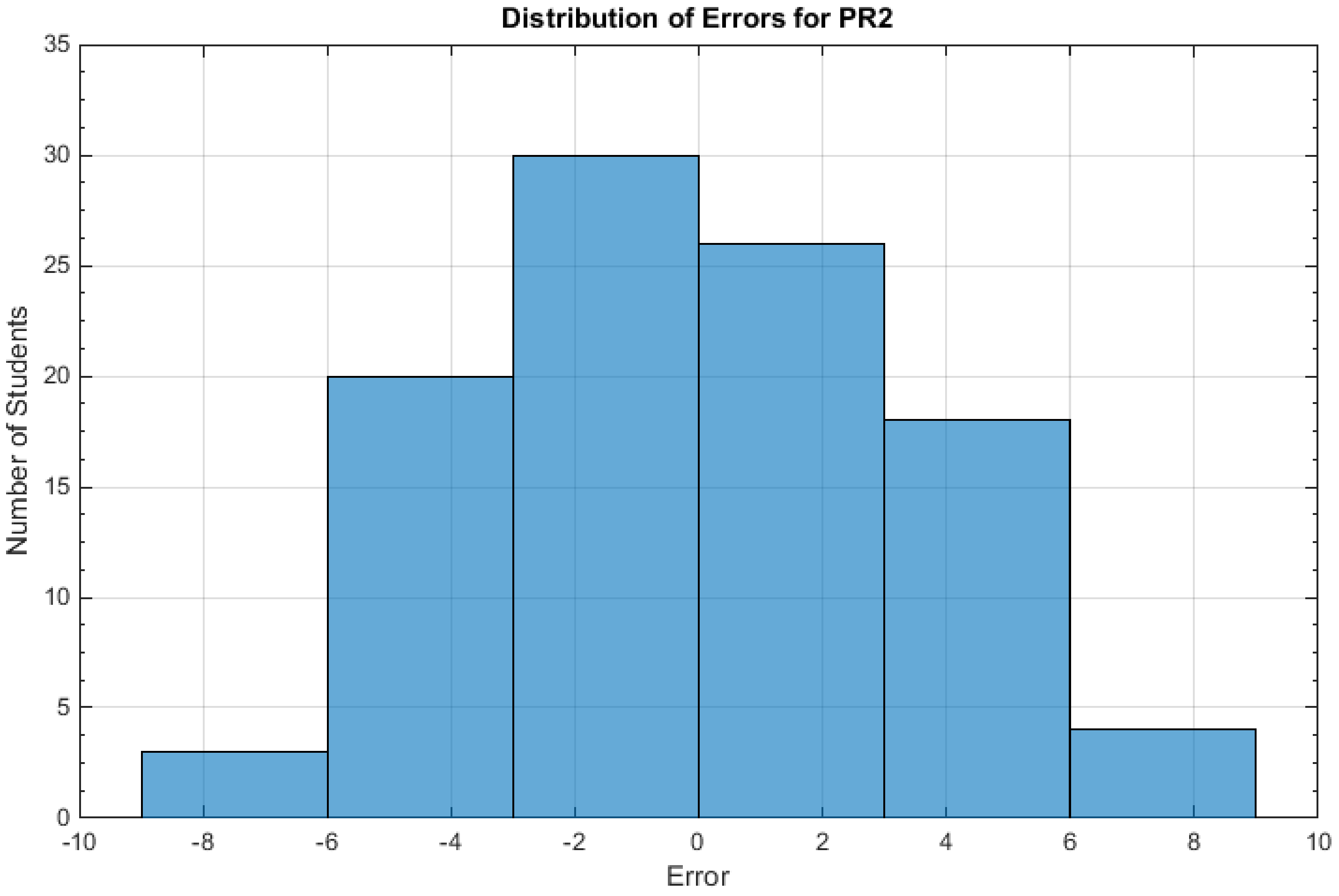


Figure 5.4: Histogram of the prediction errors for PR2



Predicting grades on a wide spectrum, from 18 to 30 in this case, is a fine-grained prediction approach in which an attempt is made to pinpoint actual student scores. When it comes to assessing the performance categories of students, it would benefit to group scores into ranges, which is usually common in grading systems used by many educational institutions such as those in the US. Such grading systems usually identify students as belonging to a number of performance categories such as poor, low, average, above average or high performers. Predicting whether grades would fall in one of these categories would provide a much better information for teachers who seek to identify and closely supervise low performing and at-risk students.

Hence, a follow-up experiment was conducted in order to explore whether a similar model with high predictive performance could be built by transforming the dataset to reflect performance categories instead of actual scores.

## 5.4 From Predicting Scores to Predicting Grades

In order to transform actual scores into grades, a scaling rule was applied to divide the scores into five ranges much similar to the A to F grading system. The only difference was that numerical grades were used instead of letter grades.

Hence, scores 28 to 30 were assigned a grade of 4, 25 to 27 a grade of 3, 22 to 24 a grade of 2, 18 to 21 a grade of 1 and those below 18 a grade of 0.

In order to perform this experiment, the same features explained earlier were used to build a new model that predicted numerical grades using the newly transformed data. As before, 10-fold cross-validation was used to evaluate the performance of the model. The model that was built using data from the IG1 course performed better than that built on the PR2 course, albeit slightly. As before, the results reported here come from the model that performed better.

The prediction values of the linear regression model were continuous and did not necessarily map into one of the five grades. Therefore, a nearest-integer rounding function was applied to make the predictions valid. As a result, the RMSE of the rounded predictions, which was slightly higher than the RMSE computed for the actual predictions, is reported.

The IG1 model had a 10-fold cross-validation RMSE of 1.12, a significant decrease from the previous model's RMSE of 2.93. Again, when tested with unseen data coming from PR2, the model scored a much lower RMSE of 1.44. This was in contrast with the previous score of 3.44, for the same setting. All of the random sampling techniques discussed earlier were applied with 10000 runs in order to evaluate the performance of the new model. However, instead of sampling from kernel distributions, sampling of grades was performed using a normal distribution as the new grades resembled this distribution. The new model had consistently lower errors than any of the randomly generated samples of grades.

The prediction errors of the new model are presented in figures 5.3 and 5.4 and comparisons between the new model and the baselines are presented in table 5.3.

Due to the discrete nature of the transformed dataset, it was possible to analyze the performance of the new model in terms of accuracy. In order to demonstrate how the model's performance varied with the range predicted, the results reported in table 5.4 include its exact prediction accuracy and its within-one-grade-point and within-two-grade-point accuracies.

The new model is given by:

$$\mathrm{FS}(i) = \mathrm{C}^{\mathrm{T}}\mathrm{S}_i + 5.75$$

where S is a 14-by-n matrix (14 parameters by n students)
$\mathrm{S}_i$ is the $i^{th}$ column in S representing student $i$
$\mathrm{C}^{\mathrm{T}}$ is the transpose of the column vector C given by:

$$C = \begin{bmatrix} -0.16 \\ 0 \\ 0.06 \\ -0.02 \\ -0.01 \\ 0.05 \\ -0.47 \\ \mathbf{6.29} \\ -4.95 \\ 0.16 \\ \mathbf{2.10} \\ \mathbf{1.55} \\ -3.53 \\ -0.30 \end{bmatrix} and\, S_{\mathrm{i}} = \begin{bmatrix} TA_{\mathrm{i}} \\ TC_{\mathrm{i}} \\ QAS_{\mathrm{i}} \\ QAN_{\mathrm{i}} \\ VC_{\mathrm{i}} \\ QP_{\mathrm{i}} \\ VE_{\mathrm{i}} \\ \mathbf{VED_i} \\ VER_{\mathrm{i}} \\ VEI_{\mathrm{i}} \\ \mathbf{SQD_i} \\ \mathbf{SQR_i} \\ SQI_{\mathrm{i}} \\ NA_{\mathrm{i}} \end{bmatrix}$$

Figure 5.5: IG1 grade prediction errors

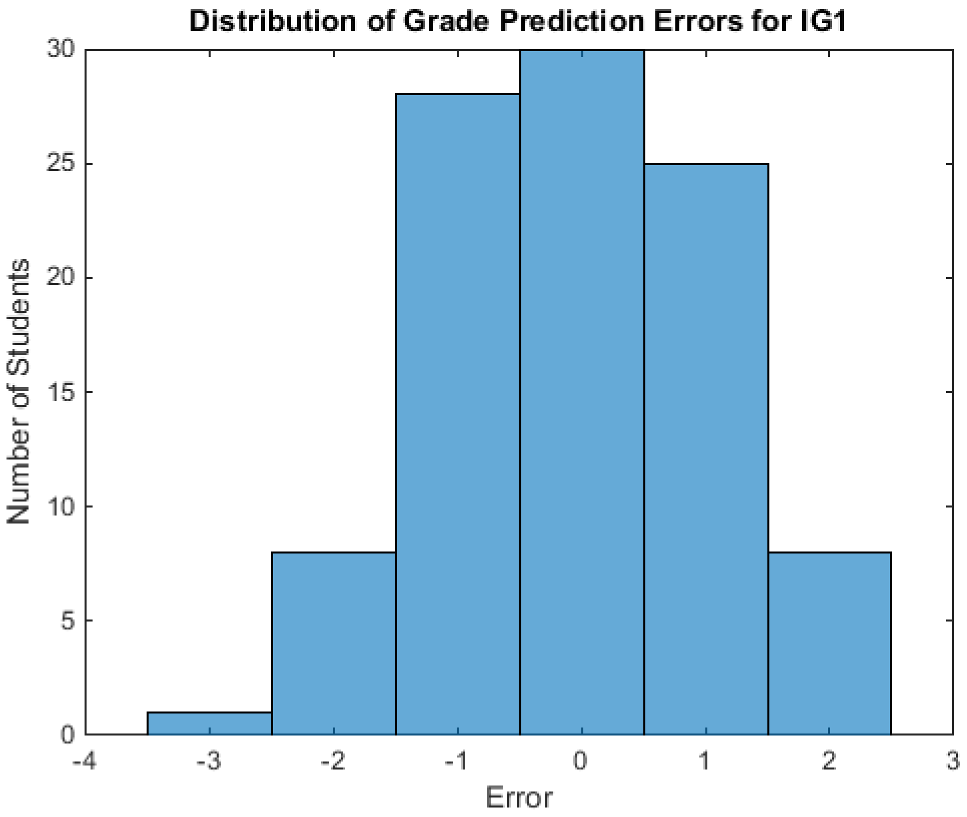


Figure 5.6: PR2 grade prediction errors

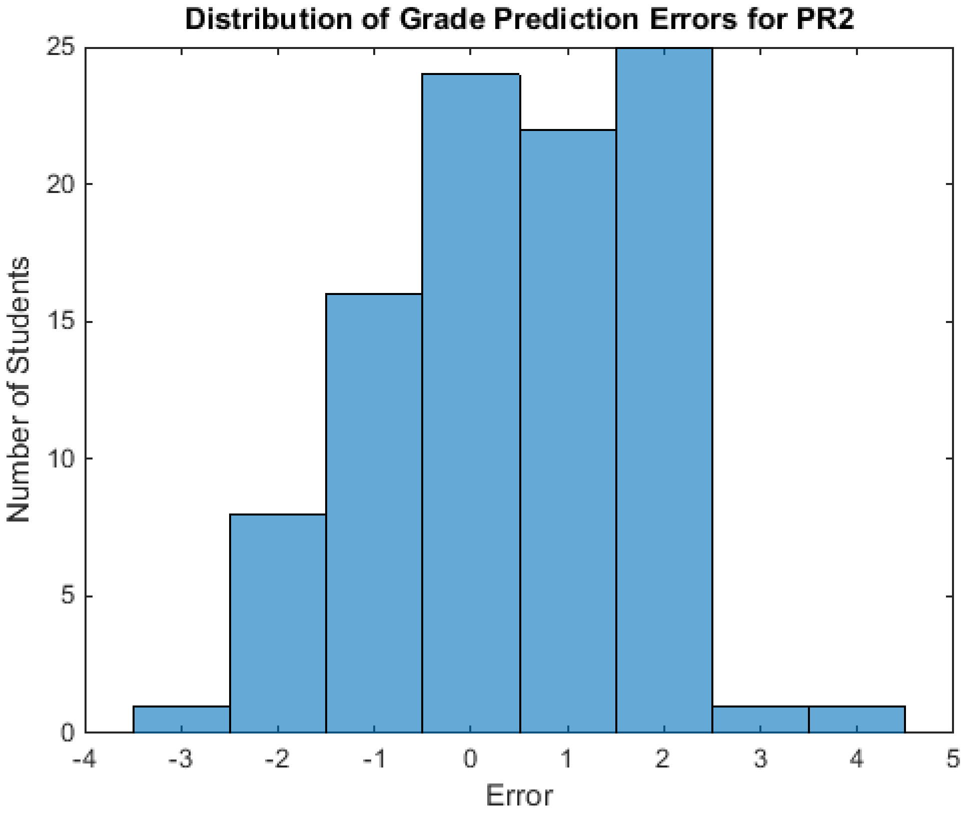


Table 5.3: Prediction errors - Root Mean Squared Error (RMSE)

| **Prediction Method** | **IG1** | **PR2** |
|---|---|---|
| Sampling from a uniform distribution | 1.83 | 1.83 |
| Sampling from a normal distribution | 1.68 | 1.65 |
| Sampling from previous scores | 1.53 | 1.53 |
| **Linear regression model** | **1.12** | **1.44** |

Table 5.4: Prediction errors - Root Mean Squared Error (RMSE)

| **Course** | **Exact** | **Within 1 Grade Point** | **Within 2 Grade Points** |
|---|---|---|---|
| IG1 | 0.30 | 0.83 | 0.99 |
| PR2 | 0.24 | 0.63 | 0.97 |

Similar to the previous model, this model rewarded students who earned votes for answering questions that were regarded as difficult and interesting, as well as for asking questions which were challenging and relevant.

The experiments showed that the new model could predict whether a grade would fall within one grade point of a prediction for over 60% of the students for both courses. This is an important observation, especially when interest is not in pinpointing actual grades of students but in the likelihood that they would perform poorly.

## 5.5 Conclusion

Performance prediction in educational activities has been studied before. Most previous studies, however, were limited to analyzing performance history of students to make such predictions. Much of this information came from high school level performance data and college entrance examination scores. as well as demographic data As discussed in the second chapter, recent studies have used current semester student data such as take-home assignments and midterms to provide such predictions.

This chapter has gone even further to demonstrate how constantly evolving data gathered from online peer-assessment activities may be used to make similar predictions. The focus was to build predictive models that would serve as a mechanism for early detection of students who might have difficulties in successfully completing their courses. This focus is an extension of previous chapters whose aim revolved around developing methodologies that promoted student engagement in order to tackle the problem of dropouts from courses, and analyzed whether peer-assessment had a positive role in fostering such engagement. Here, performance prediction using linear regression was used to demonstrate the validity of the peer-assessment methodology by proving that there is a strong correlation between participation in online peer-assessment activities and course completion. Concretely, the models discussed in this chapter showed that final scores or grades could be predicted by observing the degree of participation and success in online peer-assessment activities.

In summary, it was determined that peer-assessment data can be used as an indicator of expected student performance. It is true that the prediction models will wrongly identify some percentage of students as being at-risk. However, given the relatively smaller size of the dataset, the fact that they are strong enough to identify many of those who are indeed at risk and that all the data came from a peer-assessment system that required minimal teacher supervision makes them very promising candidates in the search for semi-supervised early intervention techniques.

One drawback of the predictive models discussed in this chapter was that they used peer-assessment data that spanned the entire duration of courses. Although all peer-assessment activities were completed well ahead of final exams, this approach may not prove to be a very practical tool of early intervention as a significant amount of time needs to pass after the course has started in order to make strong predictions.

A more reliable and timely prediction would tackle this issue by making low-error predictions in time for teachers to make the necessary arrangements to help students who may be at risk.

The next chapter demonstrates how prediction models could be built using incomplete peer-assessment data and yet provide reliable predictions as early as the first few weeks of a course to help teachers intervene in a timely manner.

# 6

# Monitoring Student Progress from Peer-Assessment Activities

## 6.1 Introduction

When used in a formative manner, peer-assessment may have much more to offer than promoting student engagement. Coupled with incentives such as some degree of contribution to final marks, it promotes voluntary student involvement.

This makes it possible to gather large amounts of online peer-assessment data. It was believed that data from such an automated environment could be used to build prediction models that would reflect expected student performance. Automation of PA tasks, together with minimal teacher intervention, would make online peer-assessment an efficient platform for gathering student activity data in a continuous manner.

It was hypothesized that, even during the early weeks of courses, participation in peer-assessment tasks could provide sound insight into student progress and that this insight could evolve with the progress of the course. The experiments conducted to examine this hypothesis adopted two different interpretations of student progress. Consequently, prediction models that explored each interpretation were built and evaluated.

Although the size of the data used in this investigation came from two courses, data from earlier versions of the courses were integrated to improve the performance of the models. It was found that, depending on the type of progress under consideration, the models could fairly trace student progress throughout both courses. In particular, encouraging results were obtained when targeting students who were likely to be at-risk of not completing their course successfully.

The findings of the previous chapter are strengthened by those discussed here and show how predictions may be applied to partial peer-assessment data so that intervention is made possible much earlier during the course.

## 6.2 Two Interpretations of Student Progress

Monitoring student progress using prediction models requires making predictions using evolving student data at several intervals. Through years of experience, teachers are usually able to make educated guesses about how students are likely to perform at end- of-course exams by studying their activities throughout the course. The goal of this study was to build prediction models that used data from previous editions of the same course in order to adopt and formalize such experience with greater efficacy.

Two interpretations of student progress were identified in relation to the peer-assessment data at hand.

One interpretation compared a student's standing at any point in the course to the standings of students at a similar point but from previous editions of the course. For example, if one collected student performance data at every week of the course, together with end-of-course grades, one would be able to compare a student's performance at any week of a course to the performance data of students from a previous edition of the course, for that specific week. If the data are good predictors of expected performance of different students in subsequent courses, one could build prediction models that would capture a teacher's expectations. Indeed, many teachers select questions that appear in exams by assessing how students from past cohorts performed on those questions. In this study, this interpretation of student progress will be referred to as ***Progress Type A***.

Another interpretation of student progress focuses on determining how far a student is from meeting course objectives. Simply stated, this evaluation may be made by comparing the expected final grade of a student at any point during a course to what is deemed to be a desirable outcome at the end of the course. This desired outcome would possibly be in the range A+ to B-. From hereon, this interpretation will be referred to as ***Progress Type B***.

## 6.3 The Data and Evaluation Metrics

Peer-assessment data collected during the course were divided into weekly data according to the three sets of tasks discussed discussed in previous chapters. The final score of each student for the course was then converted into one of four letter grades.

The data for each week incorporate the data from all previous weeks. In this manner, the prediction model for any one week is built using more performance data than its predecessors. Naturally, the data used to build the model for the first week would be modest and the data for the final week model would be complete. In general, the performances of models from consecutive weeks were expected to be better.

Similar to the models discussed in the previous chapter, the weekly prediction models were built using linear regression of the same features as before. The same performance evaluation metric, RMSE, was applied here as well. Conversion of actual scores to grades also allowed the use of other evaluation metrics.

When evaluating student performance prediction models, the two questions that are more critical than others are:

- How many of the students the model predicted not to be at-risk were actually at-risk and eventually performed poorly (False Positives) and
- How many of the students that the model predicted to be at-risk of failing were indeed at-risk (True Negatives).

  Although phrased in a different manner, technically, False Positive Rates (FPR) and and True Negative Rates (TNR) provide two interpretations of the same outcome and they are inversely proportional: $FPR = 1 - TNR$.

A prediction model with a high FPR largely fails to identify students who are at risk of failing. Conversely, a model with a high TNR identifies the majority of at-risk students. The ideal prediction model would have a very low FPR and, consequently, a very high TNR.

In order to use these metrics, the parameters used to compute them needed to be adopted to this specific problem. Hence, they were defined as follows.

- Grade – Any of the letters A, B, C, D – A and B denote high performance levels and C and D, otherwise. Although C is usually a pass grade, it is generally not favourable and was considered here as a low grade.
- Positive – A prediction that is either A or B
- Negative – A prediction that is either C or D
- True – A prediction that is either the exact outcome or falls within a one grade-point range of the actual outcome
- False – A prediction that is not True

## 6.4 Modelling Progress Type A

This type of progress monitoring compares a student's current progress at any week during the course to the progress data of past students at the same week of the course. The question that such an approach aims to answer is: 'Compared to how other students were doing at this stage in the past, how well is this student doing now?'

'How well' the student is doing is predicted as follows. First, a linear regression model is built using data collected from the first week up to the week of interest. This data comes from a previous edition of the course and the predicted variable is the final score or grade, which is already available. Then, the student's performance at the week in question is fed to the model to make a prediction. Provided that the model performs well, such weekly information shall provide insight into whether the student is likely to fall behind other students or not.

For both courses, which spanned eight weeks, a linear regression model was built per week after partitioning the complete dataset into eight respective weeks. The first week dataset contained only performance data about the first week and the final week contained contained the entire performance data. Each week contained data from the respective week plus data from all previous weeks. Hence, the comprehensive weekly dataset grew as the course progressed.

The performance of the linear regression models throughout the eight weeks was traced for each of the courses. As expected, prediction errors, measured in terms of RMSE, gradually decreased for PR2. For IG1, however, early decreases in prediction errors were followed by initial increases as the course progressed, which were followed by slight decreases towards the end of the course. In general, the decrease in errors for IG1 was not particularly satisfactory. The immediate and long-term variations in the degree of RMSE for both courses are depicted in figure 6.1.

However, RMSE measured deviations from actual scores. It would be more important for the teacher to identify performance groups than specific performance levels. Therefore, scores were transformed using the scales discussed in the previous chapter.

Yet, prediction of exact grades did not yield strong results. Specifically, high false positive rates persisted throughout the eight week period for both courses. This meant that, a significant number of students who were predicted to have obtained desirable grades did not. This observation is provided in figures 6.2 and 6.3 for PR2 and IG1, respectively.

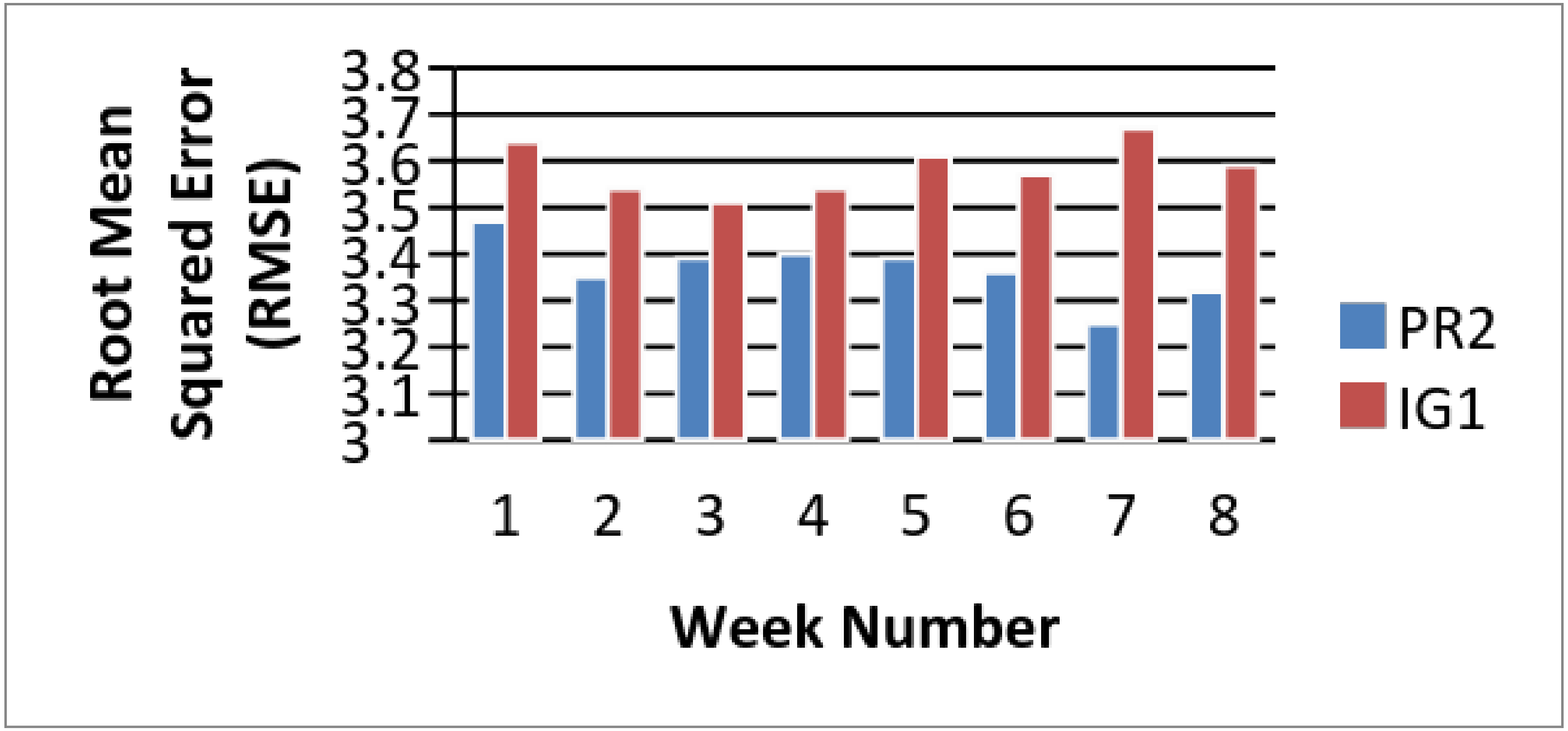
Therefore, it was decided to explore whether the prediction models performed better in identifying the performance levels of students within a one-grade-point range. Here, it was found that performance levels of the models for both courses increased significantly. Low False Positive Rates and, consequently, high True Negative Rates were recorded, even in the earliest of weeks and False Positive Rates diminished over the eight-week period.

The within-one-grade-point prediction models performed well from the very first week of the course. Although predictions were not made on exact grades, the wider range helped lower the rate of false positives and increase true positives. The same consideration may lead one to conclude that false negatives would increase, and hence, true positives would decrease. Nonetheless, the high precision and recall values for these models attested that this was not the case. The charts in figures 6.4 and 6.5 provide a visual information of the within-one-grade-point prediction performances for both courses.

The observation that the performance of the models when predicting whether a student's grade would fall within one-grade-point of the predicted value was high even at the beginning of the course and remained so throughout. This observation attested that the models could be used to identify students that may have difficulties with a course as early as the first week of the course.

The fact that the predictive models were built using data from previous editions of the course and could make predictions in a consistent manner throughout the course strongly supported the hypothesis that these models could be employed as tools of early intervention for subsequent editions of courses. Tracking student progress Type A was therefore deemed a practical application of the peer-assessment methodology.

Figure 6.1: Type A Score Prediction Errors for the models of each course over eight weeks



## 6.5 Modelling Progress Type B

The focus of this type of modeling progress is to predict how a student would perform at the end of the course by observing their current level of performance. It would be similar to asking how the student would eventually perform in final exams if their performance level did not change throughout the following weeks. This is an important question because the answer to it, if correct, would indicate whether the student has acquired all the required knowledge and skills to complete the course. Ideally, the performance levels from the first week of the course would reveal a much wider gap between current performance levels and desirable performance levels, than would those from the final weeks of the course. The ideal predictive model, would hence measure the gap between performance level at any point during the course and the desired end-of-course performance.

Modeling this type of progress only required building a single linear regression model using peer-assessment data gathered during the entire course. In order to predict a student's level of performance at any week, the student's performance levels for that week were fed to the model, which then predicted the actual score. If performance predictions from every week were concatenated to form a chain, they would provide useful information in tracking performance levels throughout the course.

The prediction performance of this model was also evaluated using several metrics. First, its exact score prediction was evaluated via RMSE. Then its performances in making exact grade as well as within-one-grade-point predictions were evaluated using the same metrics as before.

Actual score prediction errors decreased in a consistent manner from week to week, in line with the assumption that a student's expected level of performance could be predicted with higher confidence as more data about the student became available. Although this was true for both courses, IG1 prediction errors showed a much sharper decrease throughout the course.

Figure 6.2: Type A Exact grade prediction performance for PR2

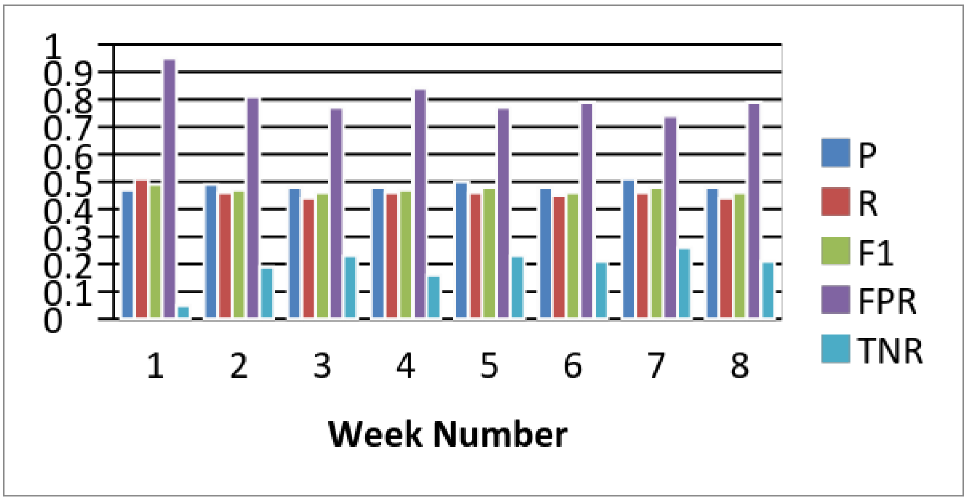


Figure 6.3: Type A Exact grade prediction performance for IG1

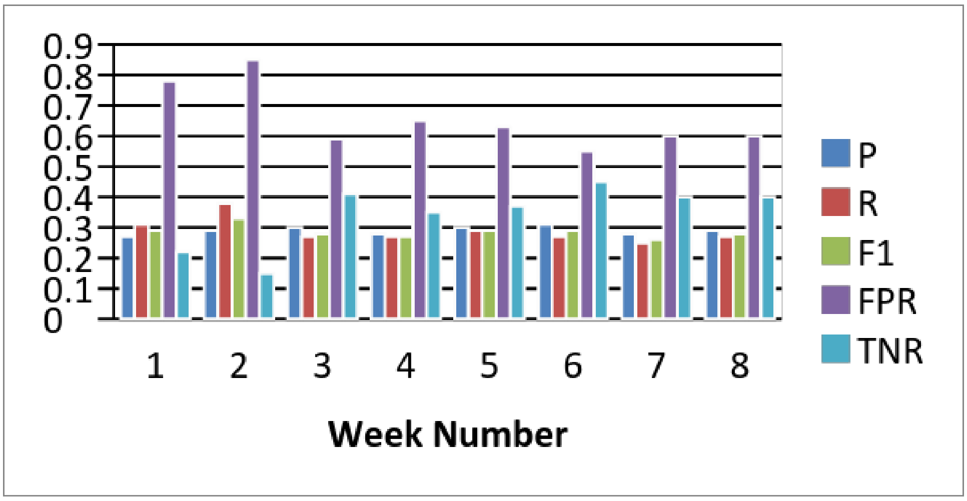


Figure 6.4: Type A Within-one-grade-point prediction performance for PR2

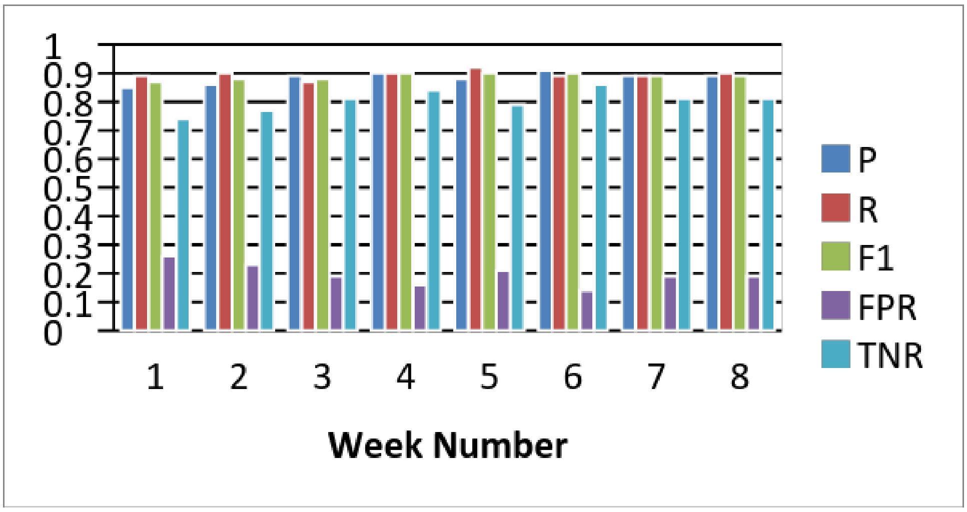
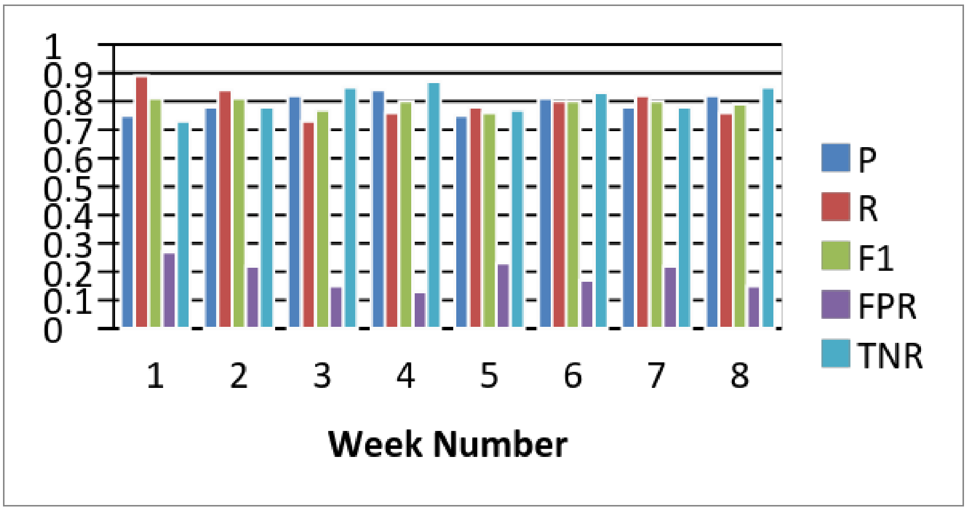


Figure 6.5: Type A Within-one-grade-point prediction performance for IG1

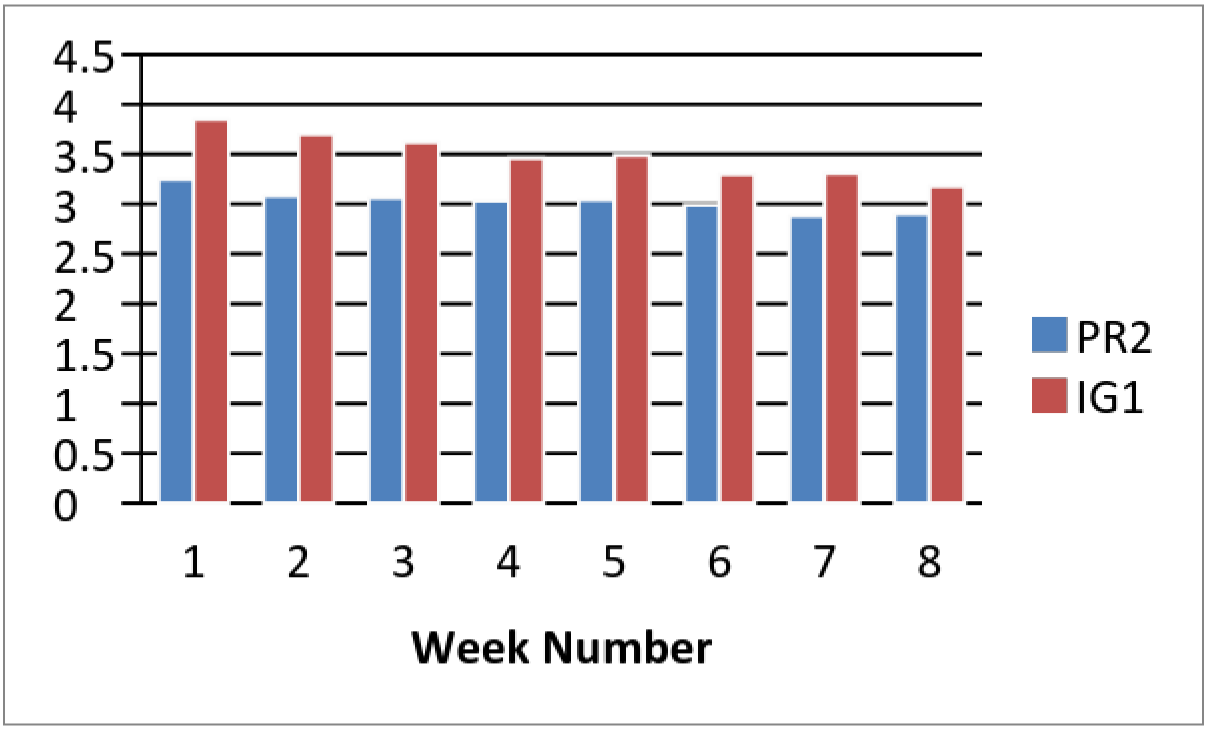


Nonetheless, prediction errors for PR2 were lower than those for IG1 for any of the weeks.

The statement that a student's end-of-course performance could be predicted with low degrees of error in the earliest weeks of the course may be construed as simplistic, or even disheartening for the student. If students knew that they were going to perform poorly in the course without even progressing through the first few weeks, they would possibly become frustrated and possibly drop out. It should be underlined, however, the aim of these prediction models is to indicate the risk of failure and that there is no harm in identifying students who may need additional assistance. Indeed, a student who finds out that they are falling behind in the course may take the appropriate measures to avoid undesired outcomes. After all, this is what early intervention is intended for.

Exact score prediction errors for both courses are depicted in figure 6.6.

Figure 6.6: Type B Prediction Errors for the models of each course over eight weeks



The second stage of evaluation involved converting actual scores into grades and constructing a prediction model using those grades. Exact grade predictions were better than those for progress type A but still low for both courses. False Positive Rates decreased throughout the weeks but were still not satisfactorily low.

Exact grade prediction performances for both courses are depicted in figures 6.7 and 6.8.

Figure 6.7: Type B Exact grade prediction performance for PR2

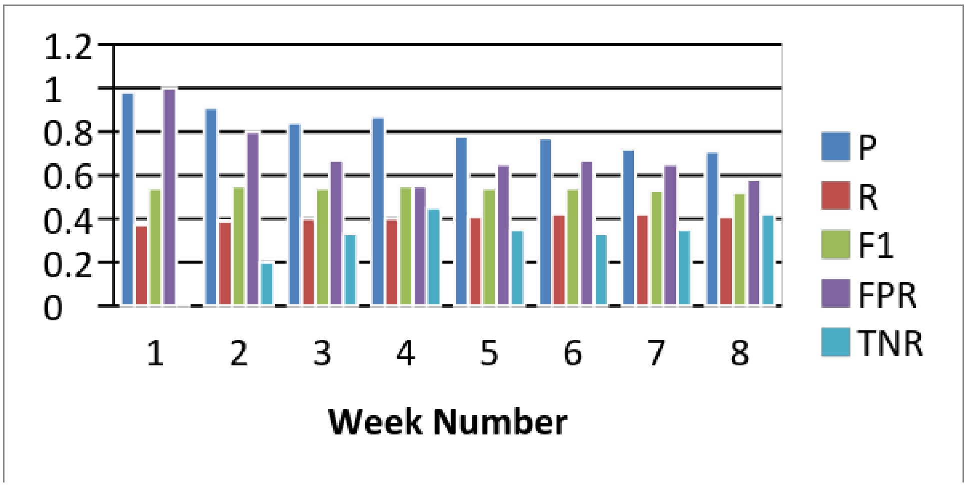
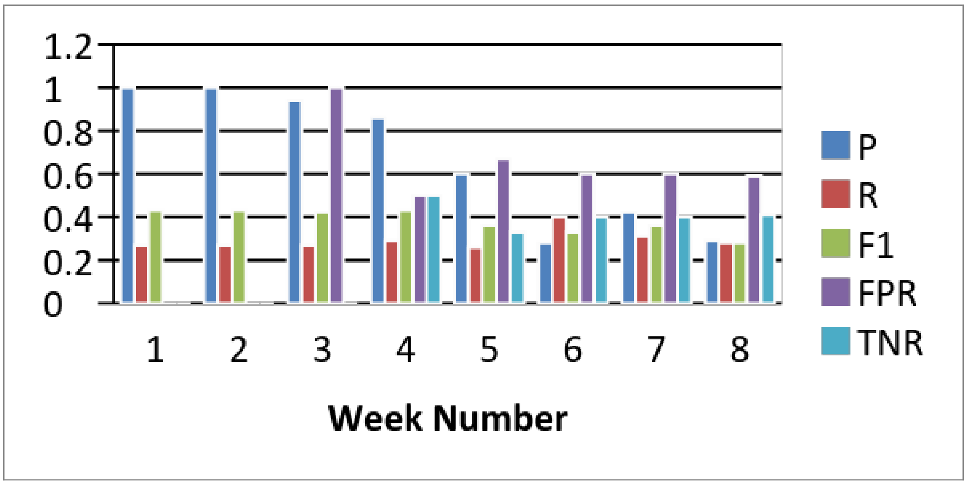


Figure 6.8: Type B Exact grade prediction performance for IG1



The next phase of the experiment studied how the model performed in making predictions that lied at a one-grade-point distance from actual grades. Similar to the models of progress type A, this model had very high levels of performance when making such predictions. Indeed, high prediction performance was obtained as early as the first week of courses. Throughout the eight weeks, False Positive Rates were either nonexistent or significantly low for both courses. This implied that a student's expected end-of-course performance could be predicted, within-a-grade-point distance, as early as the first week of the course by observing their performance in online peer-assessment activities.

Within-one-grade-point grade prediction performances for both courses are depicted in figures 6.9 and 6.10.

Figure 6.9: Type B Within-one-grade-point prediction performance for PR2

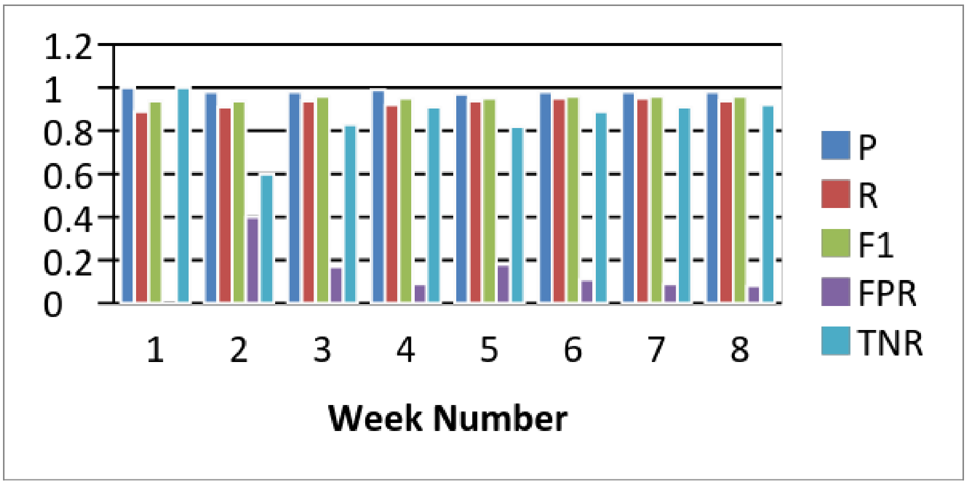
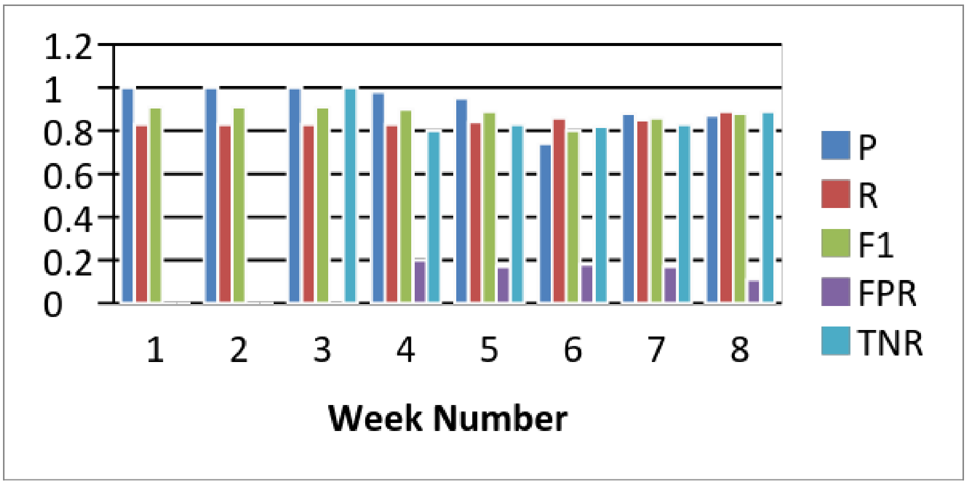


Figure 6.10: Type B Within-one-grade-point prediction performance for IG1



## 6.6 Conclusion

From peer-assessment tasks that were conducted over an eight-week period in two courses, data were used to build several prediction models according to two distinct interpretations of performance. The first interpretation focused on comparing the performance of a student at any week during the course to performance data of past students in the equivalent week. The second focused on measuring how far a student was from achieving the desired level of performance at the end of the course.

Although exact grade predictions did not produce satisfactory levels of performance for either approach, high levels of performance were observed for both interpretations of student progress when making within-one-grade-point predictions. This observation highlighted the promising potential of the peer-assessment methodology to serve as a tool of early intervention.

The similar performance of the models across the courses IG1 and PR2, especially in making within-one-grade-point predictions, demonstrated that the results were not anecdotal but consistent across both courses.

# 7

# Estimation of Student Proficiency and Test Item Difficulty from Peer-Assessment Data

## 7.1 Introduction

Pedagogical roles of the peer-assessment methodology and the models that can be constructed on top of it have been discussed so far. The peer-assessment data provide another opportunity to explore whether online peer-assessment may serve yet another purpose with respect to automated question selection and measurement of student proficiency. These potentials are explored in detail in this chapter.

Peer-assessment is one of many educational practices that stand to benefit from the automation of learning and assessment activities. Digital traces of student activities carry much information. How to predict student performance using such information obtained from a peer-assessment experiment was explored previously. This chapter discusses how peer-assessment data may help build models that predict student performance on individual test items. The semi-automated peer-assessment platform discussed in previous chapters was once again utilized in a course involving over 170 students of a first-year undergraduate computer programming course.

The main goal of the experiments discussed in this chapter is to determine if peer-assessment has the potential to be included in the category of student performance predictors that are obtained from both traditional student activity data such as performance on home works and midterms and activity on online educational platforms.

The practice of estimating student proficiency on specific test items hardly uses data from peer-assessment activities. For this reason, it should be noted that, at this stage, it is too early to delve into a discussion of the relative performance of the approach discussed here. Yet, an initial comparison of performance with a similar approach is provided.

Item Response Theory (IRT) is a well-researched topic in psychometrics and has wide applications in standardized testing, Computer Adaptive Testing (CAT) and Intelligent Tutoring Systems (ITS). IRT, in general, seeks to model the probability of a specific response by a person to an item [40]. Items may be dichotomous, with two possible responses, or polytomous, with more than two possible responses.

One of the simplest IRT models is a 1-parameter IRT logistic model, which estimates the probability of a correct response to an item with a certain level of difficulty, given a student's ability. There are many other variations such as the 2-parameter IRT model, which also takes into account the test item's discriminative power. The 3-parameter IRT model is more sophisticated as it includes a guessing parameter. Depending on whether responses are dichotomous or polytomous, many variations of IRT exist. The general representation of the IRT models with one, two and three parameters is:

$$p(u_{ij} = 1 \mid \theta_i) = c_i + \frac{1 - c_i}{1 + exp(-a_j(\theta_i - b_j))} \tag{7.1}$$

where $p(u_{ij} = 1 \mid \theta_i)$ represents the probability $p$ that the response $u$ of a student $i$ with ability $\theta_i$ is correct, $c_i$ is a guessing parameter, $a_j$ is a parameter that models how good item $j$ is at discriminating between students with close ability levels and $b_j$ is the difficulty level $b$ of item $j$.

The 1-parameter IRT model is a special case of the model where the pseudo-guessing parameter does not exist ($c = 0$) and the item's discriminative power is fixed at 1. That is, the 1-parameter IRT model is given by the formula:

$$p(u_{ij} = 1 \mid \theta_i) = \frac{1}{1 + exp(b_j - \theta_i)} \tag{7.2}$$

The Item Characteristic Curve (ICC) of an IRT model explains the relationship between item difficulty and student ability graphically. An ICC with a theoretically modeled question is shown in figure 7.1. The $x-Axis$ of the ICC represents the range of abilities. Although, technically, ability values may have a much wider range, values outside the range $-4$ to $+4$ are rarely of interest. The $y-Axis$ of the ICC represents probabilities of a correct response.

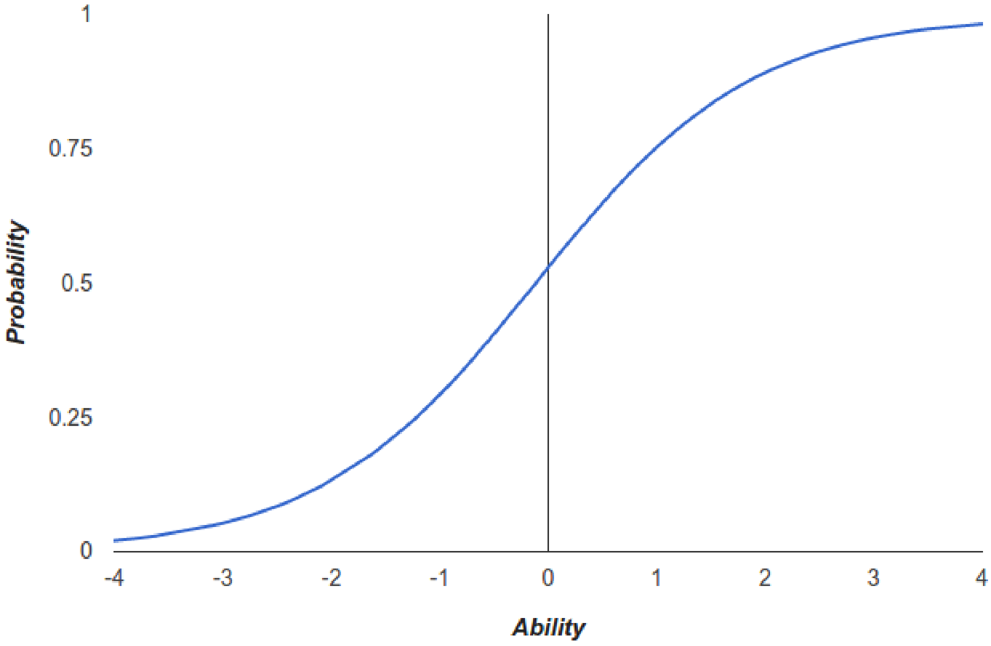


Figure 7.1: A hypothetical Item Characteristic Curve

The ICC traces a question's level of difficulty across these abilities and the probabilities with which students of varying ability levels can provide a correct response to a question. A question such as the one shown in figure 7.1 is considered to be neither easy nor difficult because a student with an ability of 0 has a 50% chance of providing a correct response to it.

Research that applies IRT to peer-assessment is modest. Ueno and Okamoto [91] explored how to improve accuracy of peer-assigned scores using a graded response model that considered a rater-characteristics parameter. Some studies that implemented IRT in other e-learning environments used several methods to model student abilities and item difficulties. Chen et al. [18] used a 1-parameter IRT model and applied a collaborative voting approach to model item difficulties. To model student abilities, they applied the Maximum Likelihood Estimation (MLE) to a student's previous responses to questions and the voting-based difficulties of those questions.

Johns et al. [45] used data about 70 multiple choice items obtained from a web-based tutor and built four IRT models, two using two parameters and two others using three parameters. They also used MLE to estimate parameters. Each experiment varied in how the parameter values were selected. They were either estimated constants based on the data or drawn from a lognormal distribution. Five-fold cross-validation of the models produced accuracy levels as high as 72% and Mean Absolute Errors (MAE) and Mean Squared Errors (MSE) of 0.37 and 0.19, respectively.

The IRT model discussed here is a 1-parameter logistic function which models a student's response to an item based on the student's ability and the item's difficulty. The objective here was to model parameters from the peer-assessment data as difficulty and ability parameters so the original model, without any additional parameters, could be applied to the problem at hand.

A Linear Regression model was built for predicting performance on individual open-questions. The model had a significant bootstrapped performance of F-statistic 212 and $R^2$ .48. The model was then applied to over 40000 unlabeled instances and the predicted scores were fed into the 2-parameter IRT model. A validation framework was developed to test the performance of the IRT model using 1146 instances. Depending on several settings, the IRT model had accuracy values as high as 0.64, True Positive Rates as high as 0.67, and True Negative Rates as high as 0.87.

The results supported the argument that peer-assessment may yet extend to another dimension and highlighted its potential as an alternative psychometric instrument as well as its applicability to test item selection processes in Computerized Adaptive Testing and Intelligent Tutoring Systems.

## 7.2 A Recent Version of the Peer-Assessment System

In this study, a recent version of the semi-automated peer-assessment platform was used in a second-year introductory computer programming course. As before, the peer-assessment activities involved cycles of three tasks in which students were required to (a) submit a question about topics already discussed in class, (b) respond to a subset of the questions that were submitted and (c) rate the responses of their peers to those questions.

Unlike in previous versions, when rating responses of their peers to a question, students were assigned a specific number of points, computed using the number of answers submitted to that question. For the purpose of the peer-assessment activities, these points were referred to as coins. Students distributed these coins over the responses, where each response was assigned a range of coins between zero and five, inclusive.

In order to counter the effect of low task completion rates, the number of points earned for answering a question was reported in terms of *Effective Coins (EC)*. This number is the proportion of coins an answer was awarded. High EC values for an answer implied that the answer was found by raters to be superior to other answers and low EC values indicated otherwise.

## 7.3 The Prediction Models

At the end of the course, peer-assessment data for 172 students and student performance statistics about 272 questions were collected. Through elementary preprocessing, a dataset about how each student fared against their peers on each question they responded to was constructed. Each instance of the dataset was formed by combining statistics about a student's performance and statistics about a question's characteristics. In plain language, an instance of the dataset could be aptly rephrased as *given the performance of student **s** on other questions, **s** would earn **p** points on question **q***. In order not to introduce unintended bias into the dataset, all information about a question that would be included in the information about a student, and vice versa, were removed from each instance.

Linear regression was then applied to determine which characteristics of students and questions were good predictors of the number of effective coins earned.

The dataset for building the linear regression model included 1146 instances. A total of 12 parameters were considered, most of which had already been used for building the models discussed in previous chapters. After initial examination of the the coefficients and p-values of these parameters, only three parameters of a question and two parameters of a student were included in the final model. The parameters are as follows.

- **QTOTALSC** - The total number of points (coins) that were distributed to all answers for a question
- **QMINEC** - The minimum number of EC that was earned for answering a question.
- **QECSD** - The standard deviation of the ECs for a question.
- **STOTALEC** - The total number of ECs a student earned, not including the EC earned for this question

- **SSQAVERAGEDIFF** - The average difficulty of a question, submitted by the student and rated by peers, which the teacher selected to distribute to students for answering.

The linear regression model was validated using bootstrapping with 10000 iterations. Table 7.1 reports statistics for the linear regression parameters. Table 7.2 provides information about the overall performance of the linear regression model.

Peer-assessment tasks were designed so that questions were answered by at least four students. This, however, did not provide plenty of information about either students or questions. It was then decided to generate more information by creating a significant amount of prediction instances from the current dataset, apply 1PL-IRT to those predictions and evaluate the IRT model's performance using the current dataset.

Prediction instances were generated by pairing each student with each question. Because each student could be represented in terms of STOTALEC and SSQAVERAGEDIFF and each question could be represented in terms of QTOTALSC, QMINEC and QECSD, pairing a student and a question entailed constructing a vector of these five parameters. Those were then fed to the linear regression model to predict the outcome in terms of EC.

Combining 172 students with 272 questions in this manner resulted in 46784 prediction instances. The linear regression model was then applied to predict a student's EC for each question. The IRT model was then constructed using these predictions.

## 7.4 Experiments and Results

A common approach to assessing a student's performance on a set of questions is to determine how many of the questions the student answers correctly. Typically, a ratio of the number of correct responses to the total

Table 7.1: Coefficients of the linear regression parameters and their statistical significance

| **SECEarnedForQ** | **Observed Coefficient** | **Standard Error** | **P > \|z\|** |
|---:|:---:|:---:|:---:|
| qtotalsc | -0.002 | 0.000 | 0.00 |
| qminec | 0.617 | 0.049 | 0.00 |
| qecsd | 0.830 | 0.122 | 0.00 |
| stotalec | 0.042 | 0.004 | 0.00 |
| ssqaveragediff | 0.012 | 0.000 | 0.00 |

Table 7.2: Overall performance of the linear regression model

| **Statistic** | **Observed Coefficient** | **Standard Error** |
|---:|:---:|:---:|
| RMSE | .079 | .002 |
| F-Statistic | 212 | 22 |
| R-Squared | .481 | .026 |

number of responses is reported. Similarly, how easy a question is can be determined by observing the ratio of the number of correct responses to the question to the total number of responses to the question.

Conversely, how difficult a question is can be determined by subtracting this value (easiness) from 1 or by computing the ratio of incorrect responses to the question to the total number of responses to the question.

In typical IRT models such as the 1PL-IRT model that was applied here, no partial credits are assigned to responses. That is, a response is either correct (1) or incorrect (0). Direct application of the 1PL model to the current dataset was, however, not possible because responses were assigned a point (EC), which was continuous between 0 and 1. In order to apply the 1PL model, EC values needed to be converted into binary values. It should, nonetheless, be noted that a score of 0 in this specific setting does not necessarily imply an incorrect response or vice versa.

In order to apply the conversion, a threshold that would determine one of the binary outcomes needed to be chosen. Inspection of the standard deviations of the EC points earned for answers revealed that EC values for many responses were very close to each other. Five potential thresholds

Table 7.3: Thresholds explored for converting continuous EC values to binary outcomes

| **Threshold** | **Above Threshold** | **Below Threshold** | **Ratio** |
|---|---|---|---|
| Max EC - Min EC | 842 | 304 | 2.77 |
| Min EC + EC SD | 703 | 443 | 1.59 |
| Max EC - EC SD | 520 | 626 | 0.83 |
| Min EC * 1.2 | 767 | 379 | 2.02 |
| Max EC * 0.8 | 569 | 577 | **0.99** |

were investigated and the threshold that yielded a zero-to-one EC ratio closest to 1 was selected. The thresholds and their ratios are reported in Table 7.3. Min EC represents the minimum EC earned for answering a question, Max EC the maximum EC earned for answering a question and EC SD represents the standard deviation of ECs earned for answering a question.

In order to obtain question difficulties and student abilities, the natural logarithms of the odd ratios (logits) of responses were computed. Applying the logit helped reduce the unequal spacing that existed between responses by a student and responses to a question so that comparisons between difficulties and abilities could be made on the same scale [101]. The formulas used for computing question difficulties and student abilities were:

$$a_i = \ln(\frac{n_{correct}}{n_{total}}) \tag{7.3}$$

and

$$b_j = \ln(\frac{n_{incorrect}}{n_{total}}) \tag{7.4}$$

where

- $a_i$ is the ability $a$ of student $i$,
- $b_j$ is the difficulty $b$ of question $j$,
- $n_{correct}$ is the number of correct responses,
- $n_{incorrect}$ is the number of incorrect responses,

- $n_{total}$ is the total number of responses and

- ln is the natural logarithm.

In contrast with that of ability, the odds ratio of a question's difficulty is expressed as the ratio of incorrect responses to the total number of responses because the difficulty of a question is commonly interpreted as how many students fail to provide a correct response to it.

Once question difficulties and student abilities were computed, each question difficulty was paired with each student ability. The 1PL IRT formula was then applied to each pair to construct a matrix of probabilities. This matrix was then consulted to infer relationships between question difficulties and student abilities.

The Item Characteristic Curves for the 272 questions are presented in Figure 7.2. They depict a more realistic situation where many of the curves intersected the y-axis well above and below the 50% probability mark.

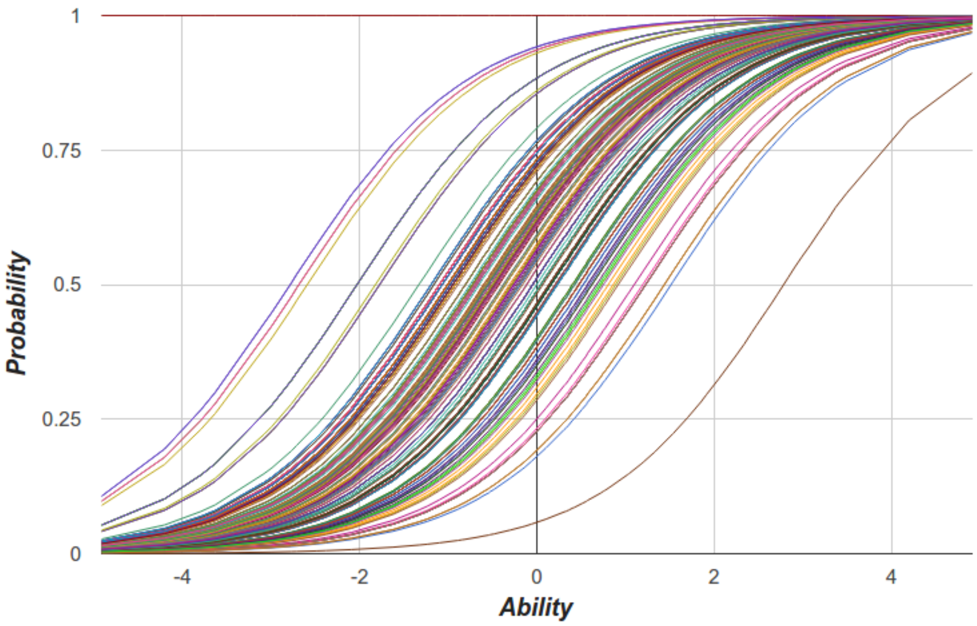


Figure 7.2: An ICC for the 272 questions in the dataset

Table 7.4 shows the estimated probabilities of scoring beyond the threshold for students with varying abilities. The IRT data suggested that a student with an ability of 0 had at least a 50% chance of scoring above the chosen threshold for about 42% of the questions. That is, for 42% of the questions, a 0-ability student was predicted to have a 50% or more chance of earning an EC that was at least 80% (the threshold) of the maximum EC earned for that question.

Table 7.4: Proportions of questions for which students of varying abilities could earn a score beyond the threshold depicted across different probability levels. *Correctness is relative to the chosen threshold.

| Ability | Proportion of questions answered correctly* with probability of at least | | | |
|---|---|---|---|---|
| | **10%** | **25%** | **50%** | **75%** |
| -3.1 | 5% | 0% | 0% | 0% |
| -1.5 | 75% | 18% | 1% | 0% |
| 0 | 100% | 89% | 42% | 5% |
| 1.4 | 100% | 100% | 94% | 61% |
| 3 | 100% | 100% | 100% | 99% |

The conversion of student performances from mere correct-incorrect ratios into logits identified 70 ability levels across the 172 students. Similarly, 69 levels of difficulty were identified for the 272 questions. The histograms in figures 7.3 and 7.4 depict the distributions for the entire range of abilities and difficulties.

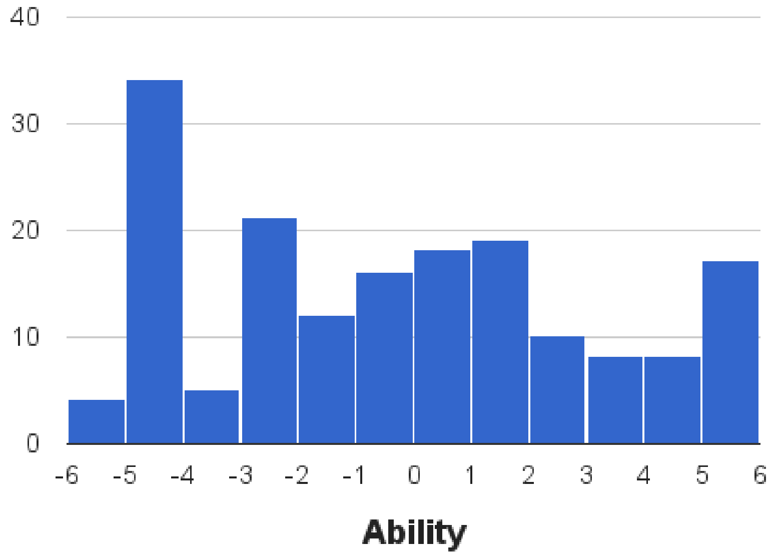


Figure 7.3: Distributions of abilities

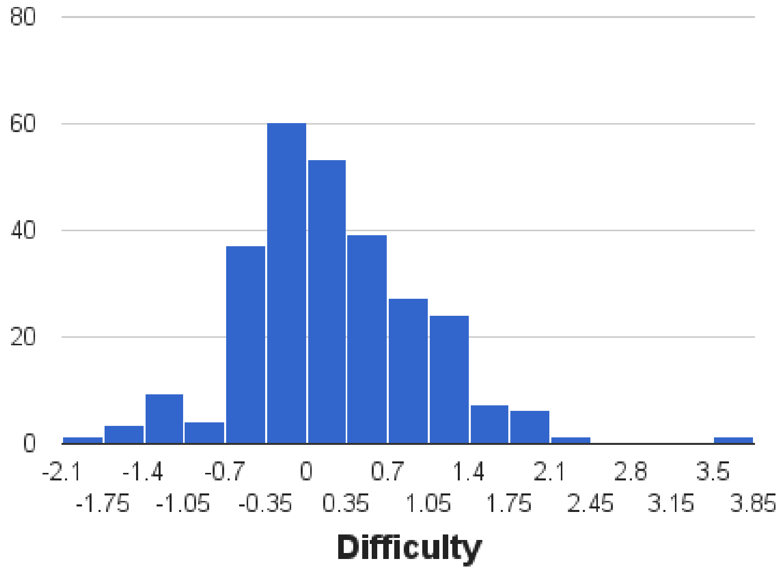


Figure 7.4: Distributions of difficulties

Abilities were spread across the entire range, with a standard deviation of 3.42. The mean ability was -0.42. There was a significant number of students with abilities far below and above average. In fact, only 56% of the students had abilities that ranged between -3 and 3, with 25% below -3 and 19% above 3.

The distribution of difficulties, however, resembled a normal distribution, with a mean of 0.22 and a standard deviation of 0.73. The large majority of the questions (69%) fell within the difficulty range -0.7 to 0.7, and almost all (93%) within the range -1.4 to 1.4. A total of 7% of the questions were outside this range, implying that they were either very easy or very difficult.

Despite the fact that the majority of the questions were close to the 0 difficulty level, they managed to identify students of varying abilities. Although the IRT model did not consider the item discrimination parameter, it could be inferred from this observation that many of the questions did exhibit significant discriminative powers. For instance, a comparison between a student with ability of 1.4 and another with ability of 3 (Table 7.4) showed that a 25% increase in probability (from 50% to 75%) was expected to reduce the proportion of "correctly answered" questions for the student with lower ability by 33%. The same increase was predicted to lead to only a 1% reduction in the expected proportion for the student with higher ability.

## 7.5 Validation of the Results

The IRT model predicted probabilities that students with varying abilities would provide "correct responses" to questions with varying levels of difficulty. Validating a model which emitted probabilities of outcomes instead of the outcomes themselves required yet another transformation of the probabilities. Hence, the model was evaluated by setting several thresholds for converting the predicted probabilities into predicted outcomes.

Table 7.5: Validation results of the IRT model across several thresholds and against two baseline predictors

| Method | Accuracy | P | R | TPR | TNR |
|---|---|---|---|---|---|
| IRT ($p >= 50\%$) | 0.64 | 0.63 | 0.67 | 0.67 | 0.61 |
| IRT ($p >= 60\%$) | 0.63 | 0.63 | 0.62 | 0.62 | 0.65 |
| IRT ($p >= 70\%$) | 0.64 | 0.65 | 0.57 | 0.57 | 0.7 |
| IRT ($p >= 80\%$) | 0.62 | 0.67 | 0.47 | 0.47 | 0.77 |
| IRT ($p >= 85\%$) | 0.62 | 0.67 | 0.44 | 0.44 | 0.79 |
| IRT ($p >= 90\%$) | 0.61 | 0.69 | 0.39 | 0.39 | 0.82 |
| IRT ($p >= 95\%$) | 0.6 | 0.71 | 0.33 | 0.33 | 0.87 |
| Always predict Correct | 0.5 | 0.5 | 1 | 1 | 0 |
| Always predict Incorrect | 0.5 | - | 0 | 0 | 1 |

In order to compare the converted values to the actual EC values, the actual EC values needed to be converted into binary outcomes as well. This, however, was a straightforward task as the same threshold chosen earlier was applied.

The 1146 instances that were used for training the linear regression model were then used to validate the IRT model. Results are reported in Table 7.5, according to the various probability thresholds that were investigated. Performances of two baseline predictors, where one always predicted a correct response and the other always predicted otherwise, are also reported. Results are reported in terms of accuracy, precision (P) and recall (R). In order to shed light on whether the predictors have a consistent performance across predicted outcomes, True Positive Rates (TPR) and True Negative Rates (TNR) are reported as well.

The results in Table 7.5 indicated that the IRT model's performance was consistent across the five evaluation metrics. A trade-off was observed between precision and recall as well as between TPR and FPR. A fair balance in performance across evaluation metrics could be struck by choosing thresholds in the range between 70% and 80%.

Extreme values of performance were recorded by the baseline predictors. Any predictor that always predicts one outcome is bound to correctly predict either all negatives or all positives, given that the ratio of actual outcomes is higher than zero. Regardless, it would fail to make correct predictions for any of the instances that belong to the other class.

Many studies that discussed how to make use of IRT in Computer Adaptive Testing (CAT) for item selection evaluated the performance of the item selection process indirectly through qualitative evaluation of the CAT module. They did not provide comparisons with other similar studies and rarely performed quantitative validation of results [72, 18, 42, 104]. Due to this and the diversity of datasets used in experiments, it was difficult to make direct comparison of IRT models in a quantitative manner.

Johns et al. [45], however, reported the accuracies of several IRT models. Although the number of questions evaluated was much less, a comparison of the highest reported accuracies and the number of questions is presented in Table 7.6.

Table 7.6: Comparison of the IRT model by [45] and the IRT model in this study

| **Study** | **Highest Reported Accuracy** | **Number of Questions** |
|---|---|---|
| [45] | 72% (experiments 1 and 2) | 70 |
| This study | 64% (experiments 1 and 3) | 272 |

## 7.6 Conclusion

Although Item Response Theory has been used in many educational environments, there are hardly any studies that utilize peer-assessment data to estimate question difficulty and student ability. The goal of this study was to extend the original goals of peer-assessment by demonstrating that it could be a useful tool in estimating test item characteristics and student abilities.

Although IRT is commonly used to estimate these characteristics in Computer Adaptive Testing environments and Intelligent Tutoring Systems, most of the data in those applications are usually gathered over long periods of time and require manually labeling correct answers. A carefully designed peer-assessment methodology has a high potential to reduce the amount of time needed to collect and label question and answer data as it distributes the load among students.

Whether student grades are valid and reliable is a topic that has been explored in great detail ever since the introduction of early forms of peer-assessment over four decades ago [27]. Although results have not been strongly positive across the hundreds of studies published since then, specific settings have been shown to lead to more valid and reliable results [27, 6]. The most significant of these is the number of assessors per task. The peer-assessment platform used in this study used a best-effort algorithm to distribute each assessment task to at least four students.

In order to assign four students to a single assessment task, the system observed the number of students and suggested how many questions should be selected from the question pool. Despite this measure, not all students completed their assigned tasks. Because the number of points distributed to answers depended on the number of assessors who completed their tasks, low task completion rates led to some answers receiving an unfairly low number of points or no points at all. In order to alleviate this problem, proportions of points earned or Effective Coins (EC) were introduced.

The entire prediction pipeline, from the linear regression model to the IRT model, relied on EC values. Due to the nature of the 1PL IRT model, thresholds had to be introduced during estimation and result validation phases.

Unlike the case of many IRT applications, the results of the IRT model should be interpreted according to the conversion threshold applied and not necessarily in terms of correct or incorrect responses. That is, a point below the specified threshold may, in reality, not imply an incorrect response. While absolute outcomes may be preferred, it is hoped that the use of thresholds will help estimate test item difficulty and student ability relative to certain points across the measurement scale.

In light of the fact that the attempt here was to demonstrate how online peer-assessment may have more to offer to the realm of predictive modeling and learning analytics, it is believed that the results of the experiments support the claim that online peer-assessment offers a source of digital traces of student activities, which may be used to build meaningful predictive models that profile both students and test items.

# 8

# Online Peer-Assessment Datasets

## 8.1 Introduction

Peer-assessment experiments were conducted among first and second year students at the University of Trento. The experiments spanned entire semesters and were conducted in five computer science courses between 2013 and 2016.

Peer-assessment tasks included question and answer submission as well as answer evaluation tasks. The peer-assessment datasets are complimented by the final scores of participating students for each course.

Teachers were involved in filtering out questions submitted by students on a weekly basis. Selected questions were then used in subsequent peer-assessment tasks. However, expert ratings are not included in the dataset. A major reason for this decision was that peer-assessment tasks were designed with minimal teacher supervision in mind. Arguments in favor of this approach are presented.

The datasets are designed in a manner that would allow their utilization in a variety of experiments. They are reported as parsable data structures that, with intermediate processing, can be molded into NLP or ML-ready datasets. Potential applications of interest include performance prediction and text similarity tasks.

## 8.2 The Datasets

Separate datasets were constructed for the five courses. For each dataset, the version of the peer-assessment system is reported as the course version. Version 1 courses used simple votes whereas version 2 courses used point-based rating of answers. Student grades have also been included in the datasets. The Italian higher education system uses a 0-30 grading scale, with 18 the minimum passing score. For students who did not complete the course successfully but participated in peer-assessment tasks, a score of 0 is reported.

It is worth stressing that peer-assessment tasks were designed with minimal teacher supervision in mind. This is evident in that none of the answers have been assigned teacher grades. While the datasets may not directly be used in peer-assessment validity experiments, they can certainly be used to build models that explore whether such validity may be inferred from course grades assigned by the teacher. Moreover, researchers may employ expert rating of answers to the datasets if they wish to carry out validity experiments. The datasets can also be readily utilized in Inter-rater reliability experiments.

The decision to make participation in peer-assessment tasks non-compulsory is reflected by the fact that some answers were assessed by a fewer number of peers than others. It is also worth noting that task incompletion rates increased towards the final weeks for all five courses. Despite this, a total of 83% of students for three of the courses completed at least a third of the tasks.

### 8.2.1 Dataset Structure

Because weekly peer-assessment tasks started with the submission of questions, a subset of which were used as inputs to subsequent tasks, it was decided to structure the datasets in a similar manner. Every course consisted of lectures, which in turn were composed of questions submitted in the first task of the week, "Ask A Question". Question attributes such as the number of evaluations and ratings of difficulty, relevance and interestingness are included. For questions that were not evaluated, values of 0 are reported.

The question text, information about the student who submitted it and, answers are also reported. Each answer structure contains the answer text, the student who provided it and its peer-ratings. Depending on the version of the course, which is also reported as a course attribute, this rating may be reported as a simple vote or as a set of coins awarded to the answer. For every student that provided an answer to a question, their course grade is reported as well.

### 8.2.2 Metadata

The complete structure of the datasets is presented in table 8.1 and an explanation for each attribute is provided in table 8.2. Over 4800 questions and over 5000 answers were submitted by more than 800 students that enrolled in the five courses between 2013 and 2016. A breakdown is provided in table 8.3.

The datasets are formatted as JavaScript Open Notation (JSON) objects. A variety of programming languages support, natively or via the use of external libraries, parsing JSON objects. A Java library that readily parses the datasets into Java objects is provided. It is hoped that, with intermediate level of processing, the JSON files can be streamlined and transformed into datasets that can be used in several machine learning and NLP tasks.

Table 8.1: Structure of the datasets

| Course | Lecture | Question | Asker | Task |
|---|---|---|---|---|
| {<br>"courseId":Integer,<br>"version":Integer,<br>"courseName": String,<br>"lectures":[**Lecture**]<br>} | {<br>“lectureId”:Integer,<br>“lectureTitle:String,<br>"questions":[**Question**]<br>} | {<br>"questionId":Integer,<br>"asker":**Asker**,<br>"task":**Task**,<br>"questionText":String,<br>"totalDifficultyLevel":Integer,<br>"totalInterestingnessLevel":Integer,<br>"totalRelevanceLevel":Integer,<br>"numEvaluators":Integer,<br>"chosenForAnswering":boolean,<br>"chosenForMultipleChoice":boolean,<br>"keywords":[**Keyword**],<br>"notes":String,<br>"answers":[**Answer**]<br>} | {<br>"courseId":Integer,<br>"askerId":Integer,<br>"courseFinalScore":Integer<br>} | {<br>"taskId":Integer,<br>"taskName":Integer<br>} |

| Keyword | Answer | Responder | Coin | Rater |
|---|---|---|---|---|
| {<br>"keyword":String<br>} | {<br>"answerId":Integer,<br>"task":**Task**,<br>"responder":**Responder**,<br>"answerText":String,<br>"notes":String,<br>"rating": Integer,<br>"coins": [**Coin**]<br>} | {<br>"courseId":Integer,<br>"reponderId":Integer,<br>"courseFinalScore":Integer<br>} | {<br>"coinId":Integer,<br>"rater":**Rater**,<br>"task":**Task**,<br>"value":Integer<br>} | {<br>"courseId":Integer,<br>"raterId":Integer,<br>"courseFinalScore":Integer<br>} |

Table 8.2: Description of dataset attributes with primitive datatypes – Integer, String and Boolean

| Name | Type | Description |
|---|---|---|
| courseId | Integer | The course's unique identifier |
| version | Integer | The version of the system used, either 1 or 2 |
| courseName | String | Name of the course |
| lectureId | Integer | The lecture's unique identifier |
| lectureTitle | String | The lecture's title |
| questionId | Integer | The question's unique identifier |
| askerId | Integer | Id of the student who asked the question |
| courseFinalScore | Integer | The student's final score for the course |
| taskId | Integer | The task's unique Identifier |
| taskName | String | The task's name, E.g. "Ask A Question" |
| questionText | String | The text of the question |
| totalDifficultyLevel | Integer | The question's difficulty as rated by students |

Table 8.3: Additional course metadata

| Course Name | Questions | Answers | Students | Dataset Filename |
|---|---|---|---|---|
| Informatica Generale 1 | 1303 | 1398 | 204 | 2_ig1.json |
| Programmazione 2 | 1013 | 1041 | 163 | 4_pr2.json |
| Programmazione 1 | 547 | 728 | 132 | 5_pr1.json |
| Linguaggi Programmazione 1 | 1087 | 1146 | 179 | 100_lp1.json |
| Lingauggi Programmazione 2 | 858 | 972 | 183 | 102_lp2.json |

The dataset files and the java parser library are freely available at https://github.com/dataset-owner/t4e_datasets. All personally identifiable information has been removed from the datasets.

## 8.3 Conclusion

Significant amounts of peer-assessment data may give back to research in the practice itself, such as large-scale validity and reliability studies as well as bring learning analytics to peer-assessment. Student performance prediction, automated essay scoring and domain specific Question Answering studies may all benefit from peer-assessment data.

The purpose of the datasets presented here is to promote such studies. None of the frameworks or studies in the literature make their peer-assessment data openly available.

The datasets contain not only information about peer-assessment experiments but also question and answer texts that may be used in Italian NLP tasks such as Question Answering and Automated Essay Scoring. Experiments using some of these datasets have demonstrated the promising potential of peer-assessment in predicting student success and modeling progress. The datasets were therefore constructed with no specific experiment in mind. However, it is hoped that their representation allows extraction of only required pieces of information with minimal effort.

One potential of use of the datasets is in research that aims to investigate the correlation between student perception of peer-assessment and performance in peer-assessment tasks. Whether participation in peer-assessment tasks contributes to successful course completion may also be investigated using these datasets.

Low task completion rate was one of the challenges faced in all rounds of peer-assessment. It influenced the completeness of the datasets and, to a degree, the fairness of points earned by students who participated in online peer-assessment tasks. Naturally, students not completing answer-rating tasks implied students whose answers were not evaluated missing out on points. Research on rating calibration to counter this effect may also be conducted using these datasets.

# 9

# Discussion and Conclusion

Peer-assessment has been practiced in classes of varying sizes for over five decades. Hundreds of studies investigating the validity, reliability and applicability of the practice at different levels of education have been conducted. Despite uncertainties that have been highlighted about its effectiveness in both formative and summative assessment environments, the practice has been widely used by institutions across the globe.

However, the majority of peer-assessment practices are conducted using traditional means of communication, namely pen-and-paper. Conducting peer-assessment in this manner requires significant time and effort, both on parts of the teacher and students. Distribution and collection of assignments reduce the efficiency and effectiveness of the practice because they take considerable amount of time and energy away from the actual task of assessment.

This undesired effect makes having peer-assessment in large classes prohibitive. Indeed, the majority of peer-assessment studies conducted in such environments deal with much smaller class sizes than would otherwise be encountered at freshman and sophomore years of college. Conducting traditional peer-assessment activities in such classes is impractical.

Perhaps the most important limitation of peer-assessment experiments conducted in such environments is the difficulty to extend those experiments to large-scale, reproducible versions. Large-scale manual peer-assessment experiments require involvement of many faculty members and, in general, require long periods of time to complete. This makes it difficult to reconstruct similar settings and reproduce experiments.

While the importance of formative peer-assessment in engaging students and helping identify at-risk students had been explored in some studies, none of those studies conducted multiple rounds of experiments with the support of online peer-assessment systems in order to explore these roles of peer-assessment in detail.

Automating peer-assessment activities, at least to a certain degree, opens the door to conducting experiments in an efficient and effective way that would be virtually impossible otherwise. Automated solutions from similar practices can be adopted, with little or no modification, to address issues that would be challenging to tackle in traditional assessment environments. Use of monitoring software and social network analysis tools, for example, could identify dishonest such as plagiarism and collusive behavior among peers.

Most importantly, automation may allow conducting large-scale experiments that may attest the role of peer-assessment in promoting student engagement as well as its potential in serving as a tool of early intervention. Automated peer-assessment data may also contain enough information to measure student proficiency and the suitability of questions that may appear in tests.

In order to explore whether these roles could indeed be served by online peer-assessment, a methodology was developed and applied in several undergraduate and postgraduate-level computer science courses at the University of Trento.

The methodology was developed with the goal of encouraging students to be more involved in class and keep up with the pace of courses, especially in classes with a large number of enrollments such as freshman and Massive

Open Online Courses. The methodology employed a mildly competitive social game where students would compete to earn a high number of points for providing answers to questions submitted by their peers.

A web-based peer-assessment system that implemented this methodology was utilized in several classes over the course of four years. End-of-course surveys of student perspectives revealed that the methodology was widely perceived as positive by students and that, according to a significantly large majority of students, it elicited increased involvement throughout.

Further analyses were conducted to determine whether there was a correlation between participation in peer-assessment tasks and successful course completion. For the three courses that were considered, a consistent relationship was found between low degrees of participation in online peer-assessment and difficulty in successfully completing courses.

These findings supported the hypothesis that the peer-assessment methodology improves student involvement and helps reduce dropout or course incompletion. They also revealed that most students who had difficulty passing final exams or successfully completing courses tended to stop participating in online peer-assessment activities during the first few weeks of courses. This supported the other hypothesis that the peer-assessment methodology serves as a tool of early intervention.

In order to explore whether the role of the methodology could be extended to identifying at-risk students before they sat final exams weeks after the completion of the courses, several prediction models were constructed using peer-assessment data gathered via the web-based platform. It was found that the prediction models could predict the final exam outcome of students with low degrees of error.

Then, it was considered that when attempting to identify students who might risk failing, teachers would be more interested in learning which performance group a student would fall in. Therefore, the efficiency of these models in making predictions within a range of grades was investigated.

The performance of the models in making such predictions improved significantly and, once again, the role of the methodology in identifying students who may require early supervision was verified.

The performance of any model that predicts events is measured not only by how good its predictions are but also by how early it can make them. Although the models discussed earlier provided low-error predictions several weeks before final exams, it was believed that true early intervention should occur at a point where there is ample time for students to adjust and for teachers to provide the necessary supervision. Hence, a later study focused on providing meaningful prediction in such a timely manner.

The focus of early identification of at-risk students shifted to weekly tracking of student progress and identification of students who might need supervision as early as halfway through a course. In doing so, two main interpretations of student progress were identified.

One compared the current performance of a student with the performances of students from previous editions of the course at the same point. This would be equivalent to asking how a student would eventually perform, judging by the outcomes of students who had a similar performance at the same stage in a previous edition of the course.

The other considered end-of-course performance of students from previous editions of the courses and sought to measure the gap between the current performance of a student and the ideal level of performance expected at the end of the course. This would be equivalent to asking how far a student is from achieving the objectives of the course.

Prediction models implementing both interpretations of student progress were thus implemented using the online peer-assessment data. Although the models did not perform particularly well when making exact score predictions on a weekly basis, they performed significantly better when predicting if student scores would fall within a range of scores. Indeed, for both interpretations of progress, a large majority of at-risk students were identified well before halfway through the course.

Therefore, continuous prediction of student progress in this manner strengthened the argument that the peer-assessment methodology could play the role of early intervention and supervision of at-risk students.

A revision of the peer-assessment methodology led to the redesign of the web-based platform to allow rating each answer for a question instead of having to choose a single answer as the best. This allowed students who provided answers to be rated consistently and led to the transformation of the peer-assessment dataset into one with much more information about each student and each question that was answered by students.

The information that was available in the recent peer-assessment dataset was deemed suitable enough to construct Item Response Theory (IRT) models. Such models rely on the argument that, given enough information about respondents and test items, it could be stated with a probability how a respondent would respond to a test item that they did not encounter before. Since the dataset was composed of information about both students and questions, such a model could be utilized, albeit with certain transformations of the dataset.

The resulting model provided information about how students would perform on questions that they had never responded to. A validation framework was developed to measure the performance of the IRT model. Although the results from these experiments were not outstanding, they were strong enough to suggest that the peer-assessment methodology, and well-designed peer-assessment practices in general, have a promising future in influencing related disciplines such as Computerized Adaptive Testing and Intelligent Tutoring Systems.

Several improvements in the peer-assessment system and the methodology are foreseen. Currently, the teacher still has to inspect and filter questions during every week of the course. This will prove to be unscalable if the number of enrollments grows significantly. Current and future efforts aim to apply advanced computing techniques such as Natural Language Processing and Machine Learning to identify and group similar questions to help with the question selection process. Another alternative is to utilize student judgments to identify and filter out those questions which are deemed inappropriate or irrelevant to the topics of the course.

For reasons that have to do beyond pedagogy, it was not possible to make peer-assessment tasks mandatory. Although it is informally accepted that not being obliged to participate in tasks may eventually affect student performance, a formal study regarding this issue is yet to be conducted. Admittedly, the non-compulsory nature of peer-assessment tasks affected the completeness of the dataset and, to a degree, the fairness of points earned by students who participated in online peer-assessment tasks.

Possible remedies to this issue include calibration of peer ratings, which also has a positive effect on the reliability of peer-assigned scores. Training students on how to assess answers has also been shown to improve their capability and experience in judging the quality of their peers' responses. This is another prospect that shall be adopted by the peer-assessment methodology as well.

Peer-assessment is a well-seasoned practice that has been utilized in all levels of education but has yet to take advantage of advances in Information and Communication Technologies. It is hoped that this thesis has demonstrated how this long-standing practice can benefit from automation of activities in furthering its adoption and advancing research into how it may evolve as a technology-supported educational discipline.

# Appendix A

# Evaluation of Selected Prediction Studies

After analysis of the studies, seven parameters that most of the studies had reported were chosen to provide comparisons. These parameters are reported in the following table. Remarks regarding whether predictions are one-off, continuous or have to potential to be transformed into continuous predictions are also provided. Where possible, F1 scores have been calculated using precision and recall.

## Acronyms

**CS** – Computer Science
**BN** – Bayesian Networks
**IBL** – Instance-Based Learning
**RF** – Random Forest
**SVM** – Support Vector Machines
**MN** – Markov Networks
**MSE** – Mean Squared Error
**TP** – True Positive
**FN** – False Negative
**FNR** – False Negative Rate

**NB** – Naïve Bayes
**NN** – Neural Networks
**DTR** – Decision Trees
**LiR** – Linear Regression
**PCA** – Principal Component Analysis
**PLS** – Partial Least Squares
**RMSE** – Root Mean Squared Error
**TN** – True Negative
**TPR** – True Positive Rate
**P** – Precision

**NBC** – Naïve Bayes Classifier
**DR** – Decision Rules
**DTA** – Decision Tables
**LoR** – Logistic Regression
**LDA** – Latent Dirichlet Allocation
**MAE** – Mean Absolute Error
**AAPE** – Average Absolute Prediction Error
**FP** – False Positive
**FPR** – False Positive Rate
**R** – Recall

| Study | Course Level & Discipline | Student and/or Data Size | Predictor Details | Outcome and Type | Algorithm or Technique | Evaluation Metric | Performance | Remarks |
|---|---|---|---|---|---|---|---|---|
| [32] | Undergrad CS | 153 students | Partial & Final marks | Final Marks | NN | not reported | not reported | one-off |
| [62] | Undergrad physics | 227 records | 184 problems online success rate | Final grades | BN, NN, DTR | Accuracy | <= 82.3% | one-off |
| [3] | not at course level | 101 records | Secondary school marks + college entry test | college performance | NN | accuracy | mean acc. 85% | one-off |
| [53] | Grad CS | 354 records | Scores on four written assignments four optional face-to-face meetings with tutors | Final mark | m5, NN LiR, SVM | MAE | 1.23 - 1.83 | could be used to provide continuous predictions |
| [35] | undergrad CS | 85 students | Prev. academic performance + demographic data | Cumulative GPA | LiR | Correlation (R) | 0.052-0.1 | one-off |
| [66] | undergrad and grad | 21428 records | academic + demographic data | pass/fail | DTR, BN | accuracy | 93-94% and 71-73% resp. | one-off |
| [68] | business course level not provided | 1360 records | academic data | performance classification | multiple | accuracy | mean acc. 97.3% | one-off |
| [47] | not reported | 1407 records | demographic data | graduation | NN | MSE, accuracy | 0.22 and 68.28% resp. | one-off |
| [4] | medical sciences | 306 records | academic + demographic data | GPA | NN | MSE | mean MSE .48 | one-off |
| [55] | grad, course and level not reported | 60 records | prev. semester marks | 5th semester marks | DTA, DTR | TPR, FPR | calculated F1 0.94 | one-off |

| Study | Course Level & Discipline | Student and/or Data Size | Predictor Details | Outcome and Type | Algorithm or Technique | Evaluation Metric | Performance | Remarks |
|---|---|---|---|---|---|---|---|---|
| [88] | undergrad not at course level | 52 records | high school+ prev. semester data | exam results | LiR-based pass/fail | accuracy | 77.8% | can be used to provide progress prediction |
| [5] | undergrad engineering | 392 matriculation and 505 diploma students | previous course results | GPA | NN | correlation, MSE | R<=0.98<br>MSE<=0.05 | one-off |
| [9] | undergrad CS | 2427 records | courses taken | degree of achievement | multiple | accuracy | 89.5%-94.9% | one-off |
| [31] | undergrad engineering | 6584 records | socio-demographic data + diagnostic test results | performance-level classification | NBC | accuracy | 60% | one-off |
| [42] | undergrad engineering | 2151 records<br>239 students | performance in other courses + partial scores | final score | LiR, NN, SVM | accuracy | SVM 89%-91% | could provide progress prediction |
| [77] | undergrad math & physics | 1540 records | SAT + pre-enrolment tests | student risk level | RF | accuracy | 90.5% | one-off |
| [16] | five CS courses different levels | 2994 records | demographic + academic data | grade | SVM, NBC, IBL, DR, DT | accuracy | 31.4%-42.2% | one-off |
| [43] | undergrad physics | 302 records | online activity + midterm | final exam | PCA + PLS | RMSE | 0-0.9 PLS,<br>0-1.5 PCA | can be used to provide progress prediction |
| [71] | undergrad not course specific | 5955 records | demographic + academic data | dropout | NN | accuracy | 75% | one-off |
| [92] | CS undergrad? | 66000 records<br>152 students | online activity | pass/fail/ excellent | BN | accuracy | 78% | can be used to provide continuous predictions |

| **Study** | **Course Level & Discipline** | **Student and/or Data Size** | **Predictor Details** | **Outcome and Type** | **Algorithm or Technique** | **Evaluation Metric** | **Performance** | **Remarks** |
|---|---|---|---|---|---|---|---|---|
| [94] | CS undergrad? | 45 records | exercises | marks & grades | LiR & custom algorithm | RMSE, accuracy | RMSE 6.9%<br>accuracy 75% | continuous prediction |
| [97] | arts, math, business | over 5 million records + demographic data | online activity | pass/fail | SVM | F1 | <=0.49 | can be used to provide continuous predictions |
| [102] | not reported | 168 records | online activity | grade | NN | accuracy | <=90% | can be used to provide continuous predictions |
| [2] | undergrad engineering | 429 students | academic + demographic data + online activity | pass/fail | NB, RF, DTR, LoR | accuracy | 0.97-0.98 | could be used to provide continuous predictions |
| [10] | not reported | approx. 30000 students | online activity | inactivity & dropout | SVM | accuracy, Cohen's Kappa, P, R, F1 | 80.4%, 0.07, 0.06, 0.1 resp. | could be used to provide continuous predictions |
| [19] | undergrad & grad not at course level | 2687 records | demographic + academic data | dropout | DTR | accuracy | 61.6%-81.5% | one-off |
| [29] | undergrad CS | 56 records | assignments | achievement of learning objectives | SVM | P,R | calculated F1 0.86-0.98 | continuous prediction |
| [37] | undergrad CS | 627 records | tests + academic year + gender | pass/at-risk/fail | DTR | P, R, F1 | mean F1 0.88 | can be used to provide continuous prediction |
| [74] | undergrad biology | 37,933 records | first week assignment completion + online activity + academic background | completion & type of certificate | LoR | accuracy, P, R, ROC area, F1 | F1 positive 0.85<br>F1 negative 0.95<br>for completion<br>F1 positive 0.79<br>F1 negative 0.8<br>for certificate type | can be used to provide continuous predictions |
| [59] | undergrad engineering | 1359 records | 1st semester academic data | admission to 2nd semester | NBC, NN, SVM, DTR | accuracy, TP, FP, TN, FN | calculated NBC F1 0.77-0.88 | one-off |

| Study | Course Level & Discipline | Student and/or Data Size | Predictor Details | Outcome and Type | Algorithm or Technique | Evaluation Metric | Performance | Remarks |
|---|---|---|---|---|---|---|---|---|
| [76] | undergrad not at course level | 149 records | demographic + academic data | at risk/ not at risk | NN | accuracy, P, R | 89.2%, 69%, 91% resp. | one-off |
| [78] | undergrad engineering | 300 records | demographic data + grades | grades | BN | accuracy | for three courses 70.4%, 73.1%, 35.6% | one-off |
| [79] | psychology MOOC | 1.6 million records | online activity | dropout | RF | accuracy | 88% | can be used to provide continuous predictions |
| [80] | CS undergrad? | 1273 records | online activity | pass/fail & final grade | SVM | P, R | calculated F1 1.0 for pass/fail mean F1 1.0 | can be used to provide continuous predictions |
| [86] | MOOC, type and level not reported | 14312 student logs | online activity | dropout | SVM | accuracy, Kappa, FNR | 0.69, 0.37, 0.16 resp. | can be used to provide continuous predictions |
| [82] | engineering undergrad? | 41,498 records | grades | grade | MN | MSE | 0-0.65 | continuous prediction |
| [84] | undergrad CS | 123 records | student self-evaluation comments | grade | SVM, NN | accuracy | SVM 50.7% NN 48.7% | may be used to provide continuous predictions |
| [90] | undergrad course level not reported | prev. academic data, size not reported | demographic data + high school grades | 1st year pass/fail | NBC, SVM | accuracy | NBC 65.2% SVM 73.2% | one-off |
| [7] | undergrad CS | 206 records | online activity | final score | LiR | RMSE | 2.93-3.44 | can be used to provide continuous predictions |
| [73] | psychology MOOC level not reported | over 3 million student logs | online activity | dropout | NN | accuracy, FNR, Cohen's Kappa | 0.74, 0.13, 0.43 resp. | can be used to provide continuous predictions |

| Study | Course Level & Discipline | Student and/or Data Size | Predictor Details | Outcome and Type | Algorithm or Technique | Evaluation Metric | Performance | Remarks |
|---|---|---|---|---|---|---|---|---|
| [20] | physics MOOC<br>level not reported | 37 million student logs | online activity | course completion | LDA | accuracy<br>TPR, TNR | 0.81-0.99 | continuous prediction |
| [24] | multiple courses<br>level not reported | 11,556 records | current GPA<br>grades<br>online activity | grades | collaborative multi-regression models | RMSE | 0.15 | can be used to provide continuous predictions |
| [52] | CS<br>undergrad? | 224 students | homework assignment<br>practical test<br>online activity | performance level classification | DTR | accuracy | 0.67-0.73 | continuous prediction |
| [61] | undergrad engineering | data from 700 students | homework assignment<br>midterm | grade | custom | AAPE, accuracy | AAPE 0-0.7<br>accuracy 76% | can be used to provide continuous predictions |
| [63] | undergrad not at course level | 250 students | demographic +<br>academic +<br>behavioural data | performance at end of 1st semester of 2nd year | DTR | TPR, P, R | calculated weighted F1 0.94 | one-off |
| [103] | CS<br>undergrad? | 21 million records<br>195 students | online activity | grade | NBC | accuracy<br>sensitivity<br>specificity | accuracy 0.84-0.86<br>sensitivity 0.88-0.9<br>specificity 0.48-0.65 | could be used to provide continuous predictions |

# Bibliography

[1] T. Abdel-Salam, P. Kaufftnann, and K. Williamson. A case study: do high school gpa/sat scores predict the performance of freshmen engineering students? In *Proceedings Frontiers in Education 35th Annual Conference*, pages S2E–7, Oct 2005.

[2] Everaldo Aguiar, Nitesh V. Chawla, Jay Brockman, G. Alex Ambrose, and Victoria Goodrich. Engagement vs performance: Using electronic portfolios to predict first semester engineering student retention. In *Proceedings of the Fourth International Conference on Learning Analytics And Knowledge*, LAK '14, pages 103–112, New York, NY, USA, 2014. ACM.

[3] A. S. Al-Hammadi and R. H. Milne. A neuro-fuzzy classification approach to the assessment of student performance. In *2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No.04CH37542)*, volume 2, pages 837–841 vol.2, July 2004.

[4] J. K. Alenezi, M. M. Awny, and M. M. M. Fahmy. Effectiveness of artificial neural networks in forecasting failure risk for pre-medical students. In *2009 International Conference on Computer Engineering Systems*, pages 135–138, Dec 2009.

[5] P. M. Arsad, N. Buniyamin, and J. l. A. Manan. A neural network students' performance prediction model (nnsppm). In *2013 IEEE International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*, pages 1–5, Nov 2013.

[6] M. M. Ashenafi. Peer-assessment in higher education – twenty-first century practices, challenges and the way forward. *Assessment & Evaluation in Higher Education*, 0(0):1–26, 2015.

[7] M. M. Ashenafi, G. Riccardi, and M. Ronchetti. Predicting students' final exam scores from their course activities. In *Frontiers in Education Conference (FIE), 2015. 32614 2015. IEEE*, pages 1–9, Oct 2015.

[8] M. M. Ashenafi, M. Ronchetti, and G. Riccardi. Predicting student progress using peer-assessment data. In *9th International Conference on Educational Data Mining (EDM), 2016*, July 2016.

[9] M. S. B. M. Azmi and I. H. B. M. Paris. Academic performance prediction based on voting technique. In *2011 IEEE 3rd International Conference on Communication Software and Networks*, pages 24–27, May 2011.

[10] Amnueypornsakul B., Bhat S., , and Chinprutthiwong P. Predicting attrition along the way: The uiuc model. In *EMNLP 2014*, page 55, July 2014.

[11] Stephen Balfour. Sustainable assessment: Rethinking assessment for the learning society. *Research & Practice in Assessmen*, 8(1):40, 2013.

[12] Roy Ballantyne, Karen Hughes, and Aliisa Mylonas. Developing procedures for implementing peer assessment in large classes using an action research process. *Assessment & Evaluation in Higher Education*, 27(5):427–441, 2002.

[13] Sue Bloxham and Amanda West. Understanding the rules of the game: marking peer assessment as a medium for developing students' conceptions of assessment. *Assessment & Evaluation in Higher Education*, 29(6):721–733, 2004.

[14] David Boud. Sustainable assessment: Rethinking assessment for the learning society. *Studies in Continuing Education*, 22(2):151–167, 2000.

[15] C Brewster and J Fager. Increasing student engagement and motivation: From time-on-task to homework. portland, or: Northwest regional educational laboratory., 2000.

[16] H. Bydovská and L. Popelínský. Predicting student performance in higher education. In *2013 24th International Workshop on Database and Expert Systems Applications*, pages 141–145, Aug 2013.

[17] Piech C., Huang J., Chen Z., Do C., Ng A., and Koller D. Tuned models of peer assessment in moocs. In *Educational Data Mining*, July 2013.

[18] C. Chen, H. Lee, and Y. Chen. Personalized e-learning system using item response theory. *Computers & Education*, 44(3):237–255, 2005.

[19] T. M. Christian and M. Ayub. Exploration of classification using nbtree for predicting students' performance. In *2014 International Conference on Data and Software Engineering (ICODSE)*, pages 1–6, Nov 2014.

[20] Cody A. Coleman, Daniel T. Seaton, and Isaac Chuang. Probabilistic use cases: Discovering behavioral patterns for predicting certification. In *Proceedings of the Second (2015) ACM Conference on Learning @ Scale*, L@S '15, pages 141–148, New York, NY, USA, 2015. ACM.

[21] Wood D. and F. Kurzel. Engaging students in reflective practice through a process of formative peer review and peer assessment. In *ATN Assessment Conference*, 2008.

[22] M. de Raadt, D. Lai, and R. Watson. An evaluation of electronic individual peer assessment in an introductory programming course. In *Proceedings of the Seventh Baltic Sea Conference on Computing Education Research - Volume 88*, Koli Calling '07, pages 53–64, Darlinghurst, Australia, Australia, 2007. Australian Computer Society, Inc.

[23] P. Denny, J. Hamer, A. Luxton-Reilly, and H. Purchase. Peerwise: Students sharing their multiple choice questions. In *Proceedings of the*

*Fourth International Workshop on Computing Education Research*, ICER '08, pages 51–58, New York, NY, USA, 2008. ACM.

[24] Asmaa Elbadrawy, R. Scott Studham, and George Karypis. Collaborative multi-regression models for predicting students' performance in course activities. In *Proceedings of the Fifth International Conference on Learning Analytics And Knowledge*, LAK '15, pages 103–107, New York, NY, USA, 2015. ACM.

[25] Gerald E. Evans and Mark G. Simkin. What best predicts computer proficiency? *Commun. ACM*, 32(11):1322–1327, November 1989.

[26] A. P. Fagen, C. H. Crouch, and E. Mazur. Peer instruction: Results from a range of classrooms. *The Physics Teacher*, 40(4):206–209, 2002.

[27] N. Falchikov and J. Goldfinch. Student peer assessment in higher education: A meta-analysis comparing peer and teacher marks. *Review of Educational Research*, 70(3):287–322, 2000.

[28] Nancy Falchikov. Involving students in assessment. *Psychology Learning & Teaching*, 3(2):102–108, 2004.

[29] M. Fernández-Delgado, M. Mucientes, B. Vázquez-Barreiros, and M. Lama. Learning analytics for the prediction of the educational objectives achievement. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*, pages 1–4, Oct 2014.

[30] George C. Fowler and Louis W. Glorfeld. Predicting aptitude in introductory computing: A classification model. *AEDS Journal*, 14(2):96–109, 1981.

[31] E. P. I. Garcia and P. M. Mora. Model prediction of academic performance for first year students. In *2011 10th Mexican International Conference on Artificial Intelligence*, pages 169–174, Nov 2011.

[32] T. D. Gedeon and S. Turner. Explaining student grades predicted by a neural network. In *Proceedings of 1993 International Conference*

*on Neural Networks (IJCNN-93-Nagoya, Japan)*, volume 1, pages 609–612 vol.1, Oct 1993.

[33] Edward F Gehringer. Peer grading over the web: enhancing education in design courses. *age*, 4:1, 1999.

[34] I. Goldin. Accounting for peer reviewer bias with bayesian models. In *11th International Conference on Intelligent Tutoring Systems*, 2012.

[35] P. Golding and O. Donaldson. Predicting academic performance. In *Proceedings. Frontiers in Education. 36th Annual Conference*, pages 21–26, Oct 2006.

[36] P. Golding and S. McNamarah. Predicting academic performance in the school of computing amp;amp; information technology (scit). In *Proceedings Frontiers in Education 35th Annual Conference*, pages S2H–S2H, Oct 2005.

[37] F. Grivokostopoulou, I. Perikos, and I. Hatzilygeroudis. Utilizing semantic web technologies and data mining techniques to analyze students learning and predict final performance. In *2014 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE)*, pages 488–494, Dec 2014.

[38] H. Gulati. Predictive analytics using data mining technique. In *Computing for Sustainable Global Development (INDIACom), 2015 2nd International Conference on*, page 713–716, 2015.

[39] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009.

[40] R.K. Hambleton, H. Swaminathan, and H.J. Rogers. *Fundamentals of item response theory.* Measurement methods for the social sciences series, Vol. 2. Sage Publications, Inc, Thousand Oaks, CA, US, 1991.

[41] Wynne Harlen and Mary James. Assessment and learning: differences and relationships between formative and summative

assessment. *Assessment in Education: Principles, Policy & Practice*, 4(3):365–379, 1997.

[42] S. Huang and N. Fang. Work in progress: Early prediction of students' academic performance in an introductory engineering course through different mathematical modeling techniques. In *2012 Frontiers in Education Conference Proceedings*, pages 1–2, Oct 2012.

[43] Gamulin J., Gamulin O., and Kermek D. Data mining in hybrid learning: Possibility to predict the final exam result. In *Information Communication Technology Electronics Microelectronics (MIPRO), 2013 36th International Convention on*, page 591–596, 2013.

[44] Lisa Jamba-Joyner and William F. Klostermeyer. Predictors for success in a discrete math course. *SIGCSE Bull.*, 35(2):66–69, June 2003.

[45] J. Johns, S. Mahadevan, and B. Woolf. *Intelligent Tutoring Systems: 8th International Conference, ITS 2006, Jhongli, Taiwan, June 26-30, 2006. Proceedings*, chapter Estimating Student Proficiency Using an Item Response Theory Model, pages 473–480. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[46] K. Jordan. Mooc completion rates. http://www.katyjordan.com/MOOCproject.html, 2014.

[47] S. T. Karamouzis and A. Vrettos. Sensitivity analysis of neural network parameters for identifying the factors for college student success. In *2009 WRI World Congress on Computer Science and Information Engineering*, volume 5, pages 671–675, March 2009.

[48] G. Kena, L. Musu-Gillette, J. Robinson, X. Wang, A. Rathbun, J. Zhang, S. Wilkinson-Flicker, A. Barmer, and E. Dunlop Velez. The condition of education 2016 (nces 2016-144). *U.S. Department of Education, National Center for Education Statistics.*, 0(0):1–26, 2016.

[49] Gregor Kennedy, Carleton Coffrin, Paula de Barba, and Linda Corrin. Predicting success: How learners' prior knowledge, skills

and activities predict mooc performance. In *Proceedings of the Fifth International Conference on Learning Analytics And Knowledge*, LAK '15, pages 136–140, New York, NY, USA, 2015. ACM.

[50] Peter T. Knight. Summative assessment in higher education: Practices in disarray. *Studies in Higher Education*, 27(3):275–286, 2002.

[51] Ingo Kollar and Frank Fischer. Peer assessment as collaborative learning: A cognitive perspective. *Learning and Instruction*, 20(4):344 – 348, 2010. Unravelling Peer Assessment.

[52] Irena Koprinska, Joshua Stretton, and Kalina Yacef. *Predicting Student Performance from Multiple Data Sources*, pages 678–681. Springer International Publishing, Cham, 2015.

[53] S. B. Kotsiantis and P. E. Pintelas. Predicting students marks in hellenic open university. In *Fifth IEEE International Conference on Advanced Learning Technologies (ICALT'05)*, pages 664–668, July 2005.

[54] George D. Kuh, Ty M. Cruce, Rick Shoup, Jillian Kinzie, and Robert M. Gonyea. Unmasking the effects of student engagement on first-year college grades and persistence. *The Journal of Higher Education*, 79(5):540–563, 2008.

[55] S. Anupama Kumar and M. N. Vijayalakshmi. Mining of student academic evaluation records in higher education. In *2012 International Conference on Recent Advances in Computing and Software Systems*, pages 67–70, April 2012.

[56] L. Kwok. Students' perception of peer evaluation and teachers' role in seminar discussions. *Electronic journal of foreign language teaching*, 5(1):84–97, 2008.

[57] Ngar-Fun Liu and David Carless. Peer feedback: the learning element of peer assessment. *Teaching in Higher Education*, 11(3):279–290, 2006.

[58] Kloft M., Stiehler F., Zheng Z., and Pinkwart N. Predicting mooc dropout over weeks using machine learning methods. In *EMNLP 2014*, 2014.

[59] Laci Mary Barbosa Manhães, Sérgio Manuel Serra da Cruz, and Geraldo Zimbrão. Wave: An architecture for predicting dropout in undergraduate courses using edm. In *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, SAC '14, pages 243–247, New York, NY, USA, 2014. ACM.

[60] Olliver McGarr and Amanda Marie Clifford. 'just enough to make you take it seriously': exploring students' attitudes towards peer assessment. *Higher Education*, 65(6):677–693, 2013.

[61] Y. Meier, J. Xu, O. Atan, and M. van der Schaar. Predicting grades. *IEEE Transactions on Signal Processing*, 64(4):959–972, Feb 2016.

[62] B. Minaei-Bidgoli, D. A. Kashy, G. Kortemeyer, and W. F. Punch. Predicting student performance: an application of data mining methods with an educational web-based system. In *33rd Annual Frontiers in Education, 2003. FIE 2003.*, volume 1, pages T2A–13, Nov 2003.

[63] T. Mishra, D. Kumar, and S. Gupta. Mining students' data for prediction performance. In *2014 Fourth International Conference on Advanced Computing Communication Technologies*, pages 255–262, Feb 2014.

[64] Chris Morgan and Meg O'Reilly. *Assessing open and distance learners*. Kogan Page, London, 1st edition, 1999.

[65] S. Myers. Formative & summative assessments. In *Research Starters Education*, 2014.

[66] Nguyen Thai Nghe, P. Janecek, and P. Haddawy. A comparative analysis of techniques for predicting academic performance. In *2007 37th Annual Frontiers In Education Conference - Global Engineering: Knowledge Without Borders, Opportunities Without Passports*, pages T2G–7–T2G–12, Oct 2007.

[67] OECD. Education at a glance 2012, 2012.

[68] E. N. Ogor. Student academic performance monitoring and evaluation using data mining techniques. In *Electronics, Robotics and Automotive Mechanics Conference (CERMA 2007)*, pages 354–359, Sept 2007.

[69] McLaughlin P. and N. Simpson. Peer assessment in first year university: How the students feel. *Studies in Educational Evaluation*, 30(2):135–149, 2004.

[70] D.E. Paré and S. Joordens. Peering into large lectures: examining peer and expert mark agreement using peerscholar, an online peer assessment tool. *Journal of Computer Assisted Learning*, 24(6):526–540, 2008.

[71] Mark Plagge. Using artificial neural networks to predict first-year traditional students second year retention rates. In *Proceedings of the 51st ACM Southeast Conference*, ACMSE '13, pages 17:1–17:5, New York, NY, USA, 2013. ACM.

[72] A. Rios, E. Millán, M. Trella, J. L. Pérez-De-La-Cruz, and R. Conejo. Internet based evaluation system. *Artificial Intelligence in Education: Open Learning Environments*, pages 387 – 394, 1999.

[73] Chaplot D. S., Rhim E., and Kim J. Predicting student attrition in moocs using sentiment analysis and neural networks. In *AIED 2015 Fourth Workshop on Intelligent Support for Learning in Groups*, 2015.

[74] Jiang S., Williams A., Schenke K., Warschauer M., , and O'dowd D. Predicting mooc performance with week 1 behavior. In *Educational Data Mining 2014*, 2014.

[75] Hidetoshi Saito and Tomoko Fujita. Characteristics and user acceptance of peer rating in efl writing classrooms. *Language Teaching Research*, 8(1):31–54, 2004.

[76] F. Sarker, T. Tiropanis, and H. C. Davis. Linked data, data mining and external open data for better prediction of at-risk students. In *2014 International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 652–657, Nov 2014.

[77] P. D. Schalk, D. P. Wick, P. R. Turner, and M. W. Ramsdell. Predictive assessment of student performance for early strategic guidance. In *2011 Frontiers in Education Conference (FIE)*, pages S2H–1–S2H–5, Oct 2011.

[78] A. Sharabiani, F. Karim, A. Sharabiani, M. Atanasov, and H. Darabi. An enhanced bayesian network model for prediction of students' academic performance in engineering programs. In *2014 IEEE Global Engineering Education Conference (EDUCON)*, pages 832–837, April 2014.

[79] Mike Sharkey and Robert Sanders. A process for predicting mooc attrition. In *EMNLP 2014*, 2014.

[80] M. Simjanoska, M. Gusev, and A. M. Bogdanova. Intelligent modelling for predicting students' final grades. In *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1216–1221, May 2014.

[81] B. Simon, M. Kohanfars, J. Lee, K. Tamayo, and Q. Cutts. Experience report: Peer instruction in introductory computing. In *Proceedings of the 41st ACM Technical Symposium on Computer Science Education*, SIGCSE '10, pages 341–345, New York, NY, USA, 2010. ACM.

[82] A. Slim, G. L. Heileman, J. Kozlick, and C. T. Abdallah. Employing markov networks on curriculum graphs to predict student performance. In *2014 13th International Conference on Machine Learning and Applications*, pages 415–418, Dec 2014.

[83] Dominique M.A. Sluijsmans, Saskia Brand-Gruwel, Jeroen J.G. van Merriënboer, and Theo J. Bastiaens. The training of peer assessment

skills to promote the development of reflection skills in teacher education. *Studies in Educational Evaluation*, 29(1):23 – 42, 2002.

[84] S. E. Sorour, T. Mine, K. Godaz, and S. Hirokawax. Comments data mining for evaluating student's performance. In *2014 IIAI 3rd International Conference on Advanced Applied Informatics*, pages 25–30, Aug 2014.

[85] Jan-Willem Strijbos and Dominique Sluijsmans. Unravelling peer assessment: Methodological, functional, and conceptual developments. *Learning and Instruction*, 20(4):265 – 269, 2010. Unravelling Peer Assessment.

[86] Sinha T., Li N., Jermann P., , and Dillenbourg P. Capturing "attrition intensifying" structural traits from didactic interaction sequences of mooc learners. In *EMNLP 2014*, 2014.

[87] Harm Tillema, Martijn Leenknecht, and Mien Segers. Assessing assessment quality: Criteria for quality assurance in design of (peer) assessment for learning – a review of research studies. *Studies in Educational Evaluation*, 37(1):25 – 34, 2011. Assessment for Learning.

[88] A. K. Tiwari, D. Rohatgi, A. Pandey, and A. K. Singh. Result prediction system for multi-tenant database of university. In *2013 3rd IEEE International Advance Computing Conference (IACC)*, pages 1340–1344, Feb 2013.

[89] K. Topping. Peer assessment between students in colleges and universities. *Review of Educational Research*, 68(3):249–276, 1998.

[90] B. Trstenjak and D. Đonko. Determining the impact of demographic features in predicting student success in croatia. In *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1222–1227, May 2014.

[91] M. Ueno and T. Okamoto. Item response theory for peer assessment. In *2008 Eighth IEEE International Conference on Advanced Learning Technologies*, pages 554–558, July 2008.

[92] A. Vihavainen. Predicting students' performance in an introductory programming course using data from students' own programming process. In *2013 IEEE 13th International Conference on Advanced Learning Technologies*, pages 498–499, July 2013.

[93] Rui Wang, Gabriella Harari, Peilin Hao, Xia Zhou, and Andrew T. Campbell. Smartgpa: How smartphones can assess and predict academic performance of college students. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '15, pages 295–306, New York, NY, USA, 2015. ACM.

[94] C. Watson, F. W. B. Li, and J. L. Godwin. Predicting performance in an introductory programming course by logging and analyzing student programming behavior. In *2013 IEEE 13th International Conference on Advanced Learning Technologies*, pages 319–323, July 2013.

[95] Meichun Lydia Wen and Chin-Chung Tsai. University students' perceptions of and attitudes toward (online) peer assessment. *Higher Education*, 51(1):27–44, 2006.

[96] Meichun Lydia Wen, Chin-Chung Tsai, and Chun-Yen Chang. Attitudes towards peer assessment: a comparison of the perspectives of pre-service and in-service teachers. *Innovations in Education and Teaching International*, 43(1):83–92, 2006.

[97] Annika Wolff, Zdenek Zdrahal, Andriy Nikolov, and Michal Pantucek. Improving retention: Predicting at-risk students by analysing clicking behaviour in a virtual learning environment. In *Proceedings of the Third International Conference on Learning Analytics and Knowledge*, LAK '13, pages 145–149, New York, NY, USA, 2013. ACM.

[98] Yun Xiao and Robert Lucking. The impact of two types of peer assessment on students' performance and satisfaction within a wiki environment. *The Internet and Higher Education*, 11(3-4):186–193, 2008.

[99] Adamson D. Yang D., Sinha T. and Rose C. P. Turn on, tune in, and drop out: Anticipating student dropouts in massive open online courses. In *2013 NIPS Data-Driven Education Workshop*, volume 11, page 14, 2013.

[100] C. Ye and G. Biswas. Early prediction of student dropout and performance in moocs using higher granularity temporal information. *Journal of Learning Analytics*, 1(3):169–172, 2014.

[101] C.H. Yu, A. Jannasch-Pennell, and S. DiGangi. A non-technical approach for illustrating item response theory. *Journal of Applied Testing Technology*, 9(2), 2014.

[102] Jieqiong Zheng, Zeyu Chen, and Changjun Zhou. Applying nn-based data mining to learning performance assessment. In *IEEE Conference Anthology*, pages 1–5, Jan 2013.

[103] Q. Zhou, Y. Zheng, and C. Mou. Predicting students' performance of an offline course from their online behaviors. In *2015 Fifth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, pages 70–73, April 2015.

[104] S. M. Čisar, D. Radosav, B. Markoski, R. Pinter, and P. Čisar. Computer adaptive testing of student knowledge. *Acta Polytechnica*, 7(4):139–52, 2010.