ABSTRACT

SONG, YANG. Data Sharing in Peer-Assessment Systems for Education. (Under the direction of Dr. Edward F. Gehringer).

Fifty years of research has found great potential for peer assessment as a pedagogical approach. With peer assessment, not only do students receive more copious assessments; they also learn to become assessors. In recent decades, more educational peer assessments have been facilitated by online systems. Those online systems are designed differently to suit different class settings and student groups; therefore, their designs are all different from each other: rating-based or ranking-based, reviews assigned randomly or to fixed groups, anonymous or onymous review, etc. Though there are different systems and a large number of users for each, there is a dearth of comparisons between different designs. This is mainly caused by the fact that the data generated by peer assessment systems is stored and analyzed separately; there is no standard for data sharing in this research community.

In this work, we focus on the data sharing between educational peer assessment systems. We designed a Peer-Review Markup Language (PRML) as a generic data schema to modeling and sharing data generated by different educational peer assessment systems. Based on PRML, a data warehouse can be built and different systems can ETL (Extract, Transform and Load) their data, contribute the data to the common data warehouse and share the data with other researchers.

Making use of data shared by different peer assessment systems can help researchers to answer more general research questions, e.g. are reviewers more reliable in ranking-based or rating-based peer assessment? To answer this question, we designed algorithms to evaluate assessors' reliabilities based on their rating/ranking against the global ranks of the artifacts they have reviewed. These algorithms are suitable for data from both rating-based and ranking-based peer assessment systems. The experiments were done based on more than 15,000 peer assessments from multiple peer assessment systems. We found that the assessors in ranking-based peer assessments are more reliable than the assessors in rating-based peer assessments. Further analysis also demonstrated that the assessors in ranking-based assessments tend to assess the more differentiable artifacts correctly, but there no such pattern for rating-based assessors.
Another research question that can be answered with this shared data is, how do collusions harm the peer review process? Ideally, if only a small number of students try to “game” the peer assessment process, the overall validity will not be affected much. However, one researcher found from his experience that more students became colluders through a semester – they gave each other high scores, or, even worse, gave high scores to every artifact they reviewed. In the worst case, a big number of colluders may make the honest reviewers outliers, which harms the validity of peer assessment. We have defined two different patterns of possible collusions and apply graph mining algorithms to detect the colluders in the data shared with us.
Data Sharing in Peer-Assessment Systems for Education

by
Yang Song

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Computer Science

Raleigh, North Carolina
2017

APPROVED BY:

Edward F. Gehringer
Committee Chair

Min Chi

Paul Franzon

Sarah Heckman
DEDICATION

For God gave us a spirit not of fear but of power and love and self-control.

- 2 Timothy 1:7
BIOGRAPHY

Yang Song is a member of the Expetiza team in the Department of Computer Science, North Carolina State University. Before joining NC State University, he obtained his Bachelors Degree in Computer Science in Beijing Institute of Technology of China in 2009 and Masters Degree in Computer Science in Beijing University of Posts and Telecommunications in 2012. His research passion lies in data mining, machine learning, especially in education settings.
ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Edward F. Gehringer for his guidance and patience. I would also thank my advisory committee members, Dr. Min, Chi, Dr. Paul Franzon and Dr. Sarah Heckman for their insightful comments.

I am thankful to my colleagues in Expertiza team and PeerLogic project. Special thanks to Zhewei Hu, Yifan Guo and Dr. Ferry Pramudianto for their support, including data labeling and implementation of baseline models. I thank Dr. Jennifer Kidd and Dr. Stacie I. Ringleb from Old Dominion University for trusting the Expertiza system and offering me suggestions from instructors’ perspective.

I am also grateful to Dr. Dmytro Babik, Dr. David Tinapple, Dr. Luca de Alfaro. The original design of Peer-Review Markup Language (PRML) was done jointly between them and the Expertiza team.

I would like to thank National Science Foundation (grant 1432347) for supporting PeerLogic project, which funded my PhD studies.

As always, the greatest debt I owe is my family. I would like to thank my wife, Guangxue “Panda” Xiao for her love and patience.
TABLE OF CONTENTS

LIST OF TABLES........................................................................................................................................ vii
LIST OF FIGURES........................................................................................................................................ viii
CHAPTER 1 Research Background.................................................................................................................. 1
  1.1 Peer Assessment................................................................................................................................. 1
  1.2 Online Peer Assessment..................................................................................................................... 2
  1.3 Problems.............................................................................................................................................. 4
      1.3.1 The Challenge of Data Sharing.................................................................................................. 4
      1.3.2 Questions Can Be (Better) Answered....................................................................................... 5
  1.4 The PeerLogic Project....................................................................................................................... 6
  1.5 Overview of the Approach............................................................................................................... 7
CHAPTER 2 Data Sharing Protocol for Educational Peer Assessment Data .............................................. 9
  2.1 Introduction......................................................................................................................................... 9
  2.2 Related Work.................................................................................................................................... 10
  2.3 Relational Peer Review Markup Language.................................................................................. 13
  2.4 Star Schema of Peer Review Markup Language.......................................................................... 16
  2.5 Evaluation........................................................................................................................................ 19
      2.5.1 Verification of Concepts in PRML.......................................................................................... 20
      2.5.2 Verification of Completeness............................................................................................... 22
  2.6 Contribution and Conclusion........................................................................................................... 24
CHAPTER 3 Data Warehouse for Educational Peer Assessment ................................................................. 25
  3.1 Introduction....................................................................................................................................... 25
  3.2 Related Work.................................................................................................................................... 25
  3.3 Data Sharing between Peer Assessment Systems and Data Warehouse ..................................... 28
      3.3.1 Schema Used ......................................................................................................................... 29
      3.3.2 General Design....................................................................................................................... 31
      3.3.3 Usage of ETL Tool .................................................................................................................. 32
      3.3.4 Designing ETL Applications................................................................................................. 35
  3.4 Evaluation........................................................................................................................................ 35
      3.4.1 Evaluation of ETL Process Based on Relational Schema................................................... 35
      3.4.2 Evaluation of ETL Process Based on Star Schema............................................................. 40
      3.4.3 Evaluation of Using Subsets of Star Schema in Web Services......................................... 42
  3.5 Contribution and Conclusion........................................................................................................... 50
CHAPTER 4 Comparison of Reliability of Rating vs. Ranking ................................................................. 51
  4.1 Introduction....................................................................................................................................... 51
  4.2 Related Work.................................................................................................................................... 51
  4.2.1 Rating-Based and Ranking-Based Peer Assessment............................................................. 52
  4.2.2 Rating-based Peer Assessment .................................................................................................. 52
  4.2.3 Ranking-based Peer Assessment............................................................................................... 53
  4.3 Reliability Algorithm ....................................................................................................................... 54
4.3.1 Global Ranking .............................................................................................. 54
4.3.2 Tuple-Based Reputation................................................................................. 55
4.4 Experimental Results...................................................................................... 57
4.5 Contribution and Conclusion ......................................................................... 59

CHAPTER 5 Collusion in Educational Peer Assessment ............................................. 61
5.1 Introduction ................................................................................................... 61
5.2 Related Work ............................................................................................... 62
5.2.1 Pervasive Collusion (PC) ........................................................................... 62
5.2.2 Small-circle Collusion (SCC) ..................................................................... 63
5.3 Detection of Potential Collusions .................................................................. 63
5.3.1 Dataset....................................................................................................... 65
5.3.2 Collusion Detection Algorithms ................................................................. 65
5.4 Experiment Results ....................................................................................... 70
5.4.1 Pervasive Collusion ................................................................................... 70
5.4.2 Small-circle Collusion ................................................................................ 71
5.4.3 Score Inflation Brough by Collusion ....................................................... 71
5.4.4 Rating vs. Ranking .................................................................................... 72
5.4.5 Relationship between Pervasive Collusion and Small-Circle Collusion .... 73
5.4.6 How does collusion influence reputation systems? ................................. 74
5.4.7 Will collusions increase during the semester? ........................................... 75
5.5 Contribution and Collusion .......................................................................... 76

CHAPTER 6 Summary and Future Work ................................................................. 78
6.1 Summary ...................................................................................................... 78
6.2 Further Work .............................................................................................. 80
REFERENCES ............................................................................................................ 82
# LIST OF TABLES

Table I. Comparison between our domain terminology with other potential reusable definitions ................................................................. 13
Table II. Whether there are corresponding concepts/terms in actual peer-assessment systems .................................................................................................................. 21
Table III. Additional concepts/terms in peer-assessment systems .............................................................................................................. 23
Table IV. Whether existing models/relations can be reused when ETLed into relational PRML ........................................................................................................ 39
Table IV (continued) ......................................................................................................................................................................................... 40
Table V. Whether existing models/relations can be reused when ETLed into star schema of PRML .................................................................................................................................................... 41
Table VI. Usage of PRML Entities in Different Web Services ........................................................................................................... 49
Table VII. Details about the Dataset Used in Experiment .......................................................................................................................... 57
Table VIII. Statistics of Assessor Reliabilities .................................................................................................................................................. 58
Table IX. Details about the Dataset Used in the Experiment .......................................................................................................................... 65
Table X. Comparison between Rating and Ranking on Potential Collusions ........................................................................................................ 73
LIST OF FIGURES

Figure 1. Heat bar from Mobius SLIP ................................................................. 9
Figure 2. PRML main concepts and their relations .............................................. 15
Figure 3. Data-warehouse schema ....................................................................... 18
Figure 4. Part of PRML related students and assignments ................................. 30
Figure 5. Corresponding Expertiza design of students and assignments .......... 31
Figure 6. General design of Extract, Transform and Load (ETL) process ............ 31
Figure 7. An example of Extract, Transform and Load (ETL) process .................. 32
Figure 8. Example of using ETL tool to extract and transform data .................... 33
Figure 9. Part of PRML design related to critiques, actors, and participants ........ 34
Figure 10. Corresponding Expertiza design related to critiques, actors, and participants ................................................................. 34
Figure 11. The expected n-to-n relationship between COURSE and PARTICIPANT if data is transformed into the original PRML ......................................................... 36
Figure 12. ETL process of n-to-n relation between COURSE and PARTICIPANT from Mobius SLIP to relational PRML ................................................................. 37
Figure 13. ETL process of n-to-n relation between COURSE and PARTICIPANT from CrowdGrader to relational PRML ................................................................. 38
Figure 14. An example of reputation from Hamer’s paper (2005) ......................... 43
Figure 15. Display of reviewers' reputations in Expertiza ...................................... 44
Figure 16. Display of summarized reviews for each criterion in Expertiza .......... 45
Figure 17. Display of plagiarism checker report for instructors in Expertiza ....... 46
Figure 18. Sample tile chart visualization on mock data (please note that, the color codes sentiments instead of the scores) ................................................................. 47
Figure 19. Students' user interface for bidding feature in Expertiza ...................... 48
Figure 20. Screenshot from a rating-based peer assessment system (Expertiza) .... 53
Figure 21. Screenshot from a ranking-based peer assessment system (Mobius SLIP) .... 54
Figure 22. Distribution of assessors' reliability in rating-based assignment ...... 58
Figure 23. Distribution of assessors' reliability in ranking-based assignments ....... 58
Figure 24. An example of a small-circle collusion .............................................. 68
Figure 25. Disagreement between peer-assessment scores within and outside the small-circle detected ................................................................. 68
Figure 26. An example of a pervasive colluder ................................................... 69
Figure 27. Distribution of pervasive colluders’ reviews percentages ....................... 70
Figure 28. Distribution of small-circle colluders’ reviews percentages ................. 71
Figure 29. Distribution of score inflations brought by collusions ......................... 72
Figure 30. Peer-assessment inflations based on their deadline of day of semesters... 75
CHAPTER 1 Research Background

1.1 Peer Assessment

Traditionally, feedback on student work is provided mainly by teaching staff—they mark up the hardcopies of students’ artifacts, give comments on the issues, and return the hard copies to students. This approach is generally inefficient, as teaching staff only have an opportunity to do this once for each assignment. Thus, students receive the comments and grades as final decisions. Though some formative comments might be provided, the students usually have no chance to improve their work on the current assignment.

Extensive research has shown that much more learning and skill development take place when there is bi-directional interaction in a class: Students work with both students and instructors providing feedback to the students on their artifacts (Brookhart 1999). Peer assessment is an instructional technology designed to support such interactions (K. Topping 1998).

Compared to older technologies and traditional instruction, peer assessment enables more effective and timely feedback. Student authors are informed of their strengths and weaknesses soon after a submission deadline. If a second round of submission is allowed, the authors can also use the peer assessments to improve their own work (Song et al. 2016).

Peer assessment is also reported to be beneficial to the learning process (Davies 2002; Dochy, Segers, and Sluijsmans 1999). During the peer-review phase, the reviewers can reflect on their strengths and weaknesses in their artifacts. Some research even suggests that this metacognitive process makes peer assessment more beneficial to assessors than to assesses. In 2009, Lundstrom and Bakers conducted a research on peer assessment in an intensive English institute. The students were divided into “givers” group – the group which gave peer assessments but did not receive any peer assessment – and “receivers” group – the group which received peer assessments but did not review others’ work. Then an analysis on the students writing ability showed that the givers group made “more significant gains in
their own writing” (Lundstrom and Baker 2009).

Topping theoretically suggested that peer assessment may “promote a sense of ownership, personal responsibility, and motivation” and thereby increase the students’ engagement (K. Topping 1998). Tseng and Tsai investigated peer-assessors’ motivations in an online peer-assessment done by college students. They found that students were “strongly intrinsically motivated” and consider the peer assessment to be a learning process (Tseng and Tsai 2010). Rada also found that peer assessment encourages the reviewers to improve their own work compared with the ones who did not provide peer assessment to others (Rada 1994).

1.2 Online Peer Assessment

The IT revolution has enabled a host of online peer assessment systems, which automate the workflow and remind students to complete their tasks (McLean 2013; De Alfaro and Shavlovsky 2014; Tinapple, Olson, and Sadauskas 2013; E. Gehringer 2009; Babik et al. 2016). Most of those systems share common features, as they all allow:

- instructors to set up deadlines,
- students to submit their artifacts,
- instructors to configure peer-review setups (such as review policy, review rubrics, etc.),
- students to submit peer assessments,
- instructors to view aggregated peer-assessment results.

Despite the common functionalities provided, the systems are designed differently from each other based on different class settings and objectives of the online peer-assessment systems. The designers/owners may incorporate different features, such as

- review in ad-hoc style or in closed groups,
- anonymous or onymous review,
• rating-based or ranking-based review.

In addition to the systems designed to facilitate peer assessment, Massive Open Online Course (MOOC) platforms also widely adopt peer assessment as one important feature. Popular courses on MOOC platform can have thousands or even more students. Since the majority of those courses are free, the teaching staff are stretched very thin when giving students feedback. The grading – giving summative feedback – can possibly be a burden and the formative assessment – giving insightful comments to help authors to improve, or explaining why their solutions are not correct – can hardly be done solely by teaching staff. However, if the formative assessment is missing, the learning experience will be much worse as Suen suggests: “without formative assessment and feedback, MOOCs amount to information dump or broadcasting shows” (Suen 2014). Therefore, MOOCs platforms use peer assessment as an important approach to providing formative feedbacks.

MOOCs also require a scalable means of assessment based on complex, open-ended assignments, which is another reason for those platforms to use peer assessment. Some researchers argue that peer-assessment data, plus estimating reviewers’ reliability (Lauw, Lim, and Wang 2007) and correcting reviewers’ bias (Piech et al. 2013), can provide more accurate estimates of scores for students’ artifacts. Piech et al. constructed a model to “tune” students’ peer assessment to better predict scores for all students. A small number of artifacts were assessed by the teaching staff. When assigning artifacts to reviewers, each reviewer was assigned to review one artifact which had been reviewed by the teaching staff. The researchers estimated the bias and reliability when estimating grades for each artifact. They reported that this approach was more accurate than simply assigning the median peer-review scores as the students’ grades.

Calibration in another approach which can be introduced into MOOCs (Balfour 2013). The calibration phase was introduced to peer assessment originally by UCLA’s Calibrated Peer Review™ (CPR). Before students review each other’s artifacts, there need to review some sample artifacts and calibrate their scores with instructors’ assessments. Students can learn how to make judgments better in this calibration process (Song et al. 2016).
In CPR™, the calibration phase more strongly emphasizes the training purpose: the reviewers can attempt the calibration task several times until they are able to make judgments close enough to the teaching staff’s. Some other systems have also adopted calibration to train students before they perform peer assessment on each other’s artifacts (Song et al. 2016). By contrast, the concept of calibration used in the work of Piech et al. focused on estimating the review bias and reliability for each reviewer.

1.3 Problems

1.3.1 The Challenge of Data Sharing

Despite the fact that dozens of online peer assessment systems are in use, virtually all research studies on online peer assessment derive their conclusions from data taken from a single system (Song, Hu, and Gehringer 2015a; Song et al. 2016; E. Gehringer et al. 2007; Tinapple, Olson, and Sadauskas 2013; De Alfaro and Shavlovsky 2014). A few surveys have tried to compare different systems, but they have only scratched the surface, because they compare the functionalities of the different systems, but contain no results based on the comparison of data from different systems (Luxton-Reilly 2009; Babik et al. 2016). This is symptomatic of the lack of protocols for sharing data generated by different peer assessment systems.

Different online peer assessment systems may use different terms for the same concept, or use same terms for different concepts. For example, in a peer-assessment activity, the student authors need to submit their work first. Different systems refer to this as an “artifact”, “work” or “submission”. Even the term “peer review” has different meanings in different contexts. It can mean an assessment of work submitted by a different individual or team, or it can mean a review of an individual’s contribution to the work of the team.

Moreover, different systems implement the process of peer review in different ways. For example, some systems give each reviewer several submissions and ask them to rank them; whereas other systems may give each reviewer one submission at one time and ask the
reviewer to rate it based on different aspects. There is no easy way to compare rankings from one system with ratings for another. To make sense of the results, researchers need to thoroughly understand the design of each system, and this will take a long time to achieve.

1.3.2 Questions Can Be (Better) Answered

Differences in system designs and terminologies used make it hard to compare results from different systems. However, many research questions cannot be addressed without first obtaining a sufficiently general data set.

Below are a few most important questions can be answered with a bigger data set:

**Ranking-based vs. rating-based peer assessment.** Ranking-based and rating-based are two common designs of peer assessment activities. In ranking-based peer assessment, each reviewer is assigned several artifacts and is required to give a rank for them. In rating-based peer assessment, reviewers are asked to give each artifact grades, usually one by another. Supporters of each type may argue about the advantages, e.g., ranking-based assessments are more reliable (Shah et al. 2013), rating-based assessment can support detailed rubrics better (Cho, Schunn, and Wilson 2006). A comprehensive comparison has not been done between those two peer assessment designs. In this dissertation, we provide answers on two different dimensions: students’ review reliability (Chapter 4) and students’ collusions (Chapter 5).

**Anonymous vs. "Owned" Assessments.** There is considerable debate about whether the instructors should let the reviewers know the authors of reviewed artifacts and whether the authors should know who reviewed them. On one hand, some researchers argue that reviewers can be more honest when the review is anonymous (Michaelsen, Sweet, and Parmelee 2009). On the other hand, some researchers also found that reviewers tend to be harsher when they do not "own" the peer assessments they gave (assessments are given anonymously). In the worst case, the relationships between reviewers and reviewees can be impacted as well (Lane and Trader 2007).

**Batch effect.** Researchers have observed that the score assigned to an artifact can be affected by the other artifacts reviewed by the same reviewers: the artifact tends to receive a
higher score when it is reviewed in a batch of worse artifacts, or a lower score when it is reviewed in a batch of stronger artifacts (Díez et al. 2014). This is observed by some researchers, but more validation is still needed, especially on whether detailed rubrics can help minimize the batch effect.

Those research questions mentioned above have several common features:

**Instructors may only apply one design at one time.** Instructors, when applying peer assessments to their curriculum, may not be willing to complicate the class design too much by dividing students into experiment groups and control groups. In our experience on hosting peer assessment system Expertiza, we have only seen a few cases where instructors have tried to divide their classes and apply different pedagogical interventions.

**For a single peer assessment system, not all the options are built-in.** Even if an instructor is willing to divide the students into two groups, the peer-assessment options, e.g. anonymous/“owned” assessments, may not be supported by the peer-assessment system used. It is common for the original peer-assessment system designers to make decisions according to their preferences, and building another option into an existing peer-assessment system, can be time-consuming.

**It is harder to collect comparable data set.** Every peer-assessment system is designed differently and instructors may not be able to configure the peer-assessment settings. Therefore, each peer-assessment system may hold data only for the settings supported by it. To make comparisons, a big/comparable dataset has to be collected; otherwise, the conclusions can be biased.

### 1.4 The PeerLogic Project

PeerLogic is an NSF-funded research project that provides services for educational peer-review (aka. peer-assessment) systems. The project has two main parts:
A suite of web services that provide functionality useful to peer-assessment systems, and can be used to the extent desired by the developers of those systems. Those web services can easily be incorporated into existing peer-review systems and provide useful functionalities such as reputation (Song, Hu, and Gehringer 2015b; Hu, Song, and Gehringer 2016), summarization, visualizations, etc.

A data warehouse, an open repository of anonymized data from millions of reviews performed in several different peer-assessment systems. This is intended as a resource for researchers in peer assessment that is free of the institutional bias that might be present in a corpus drawn from a single system.

Those two parts, in fact, are closely related. The web services require the client systems to model the data in a particular format, and this data format should be pre-defined by the users and designers of the web services. This data format can also be considered as a data protocol between the server and the clients. Based on different web services, the data modeled will be different. For example, for the reputation web service, a majority of the data transmitted to the web service server should be numerical, holistic scores (e.g., reviewer A gives reviewer B a 91 out of 100); for summarization web service, majority of data transmitted should be textual content to be summarized; for visualizations of detailed reviews, the web service may require both numerical and textual feedback on each criterion.

The schema of the data warehouse can be considered as a superset of all the information needed by all the different web services combined. In other words, the data protocols for those web services can be considered as subsets of the data warehouse. Therefore, the definition of a data warehouse schema is the basis of further research topics.

1.5 Overview of the Approach

In this dissertation, we address the problem of data sharing in peer-assessment systems for education. The following is the structure of this dissertation.
Chapter 2 presents our effort to design a Peer-Review Markup Language (PRML) to facilitate data sharing between systems. The Peer-Review Markup Language provides a common definition of terminology across multiple systems. Based on this PRML, different peer-assessment systems’ data can be transformed into a common PRML schema, which makes data sharing possible between different peer-assessment systems.

Chapter 3 demonstrates how a data warehouse was built based on PRML as a data-sharing schema. In this chapter, we provide discussions of the reason for using the star schema of PRML instead of the original version. We also describe the general data sharing process between client systems and the data warehouse.

Chapter 4 shows how a research question — comparison of reliability between ranking-based and rating-based systems — can be (better) answered with a data warehouse of educational peer assessment systems. Ranking-based and rating-based are two main categories of peer-assessment activity designs. We designed a tuple-based algorithm to evaluate the reviewers’ reliabilities. With the data shared by different systems, we managed to carry out a comparison on review reliability of rating-based and ranking-based reviews.

Chapter 5 discusses our findings on potential collusion of students in peer assessment. We presented our definition of two types of collusion and our algorithms to detection colluder suspects. Further comparison on the impacts from collusions on both rating-based and ranking-based peer assessment is also presented.

Chapter 6 concludes this dissertation with a summary of the work and directions for future research.
CHAPTER 2 Data Sharing Protocol for Educational Peer Assessment Data

2.1 Introduction

Peer assessment has proved to be a useful technique in all levels of education. The process of giving and receiving comments can encourage critical thinking and help students learn both from reviewing and being reviewed. Peer assessment generates a large volume of data, especially if done online.

We know of almost two dozen online systems that provide a peer-assessment feature (Luxton-Reilly 2009; Babik et al. 2016). Most of them have clear purposes and design emphases. For example, CritViz (Tinapple, Olson, and Sadauskas 2013) is a system using only ranking-based peer assessment: the reviewers only need to provide ranks to \( n \) artifacts without indicating the magnitude of quality differences. Therefore, the fact that one artifact is ranked lower than another can mean that the better one is of clearly higher quality, or that the higher one is only slightly better. MoubiusSLIP, however, asks reviewers to give ordinal assessments on a heat bar (Figure 1) on which reviewers can not only rank the reviewed artifacts, but also indicate the magnitude of the differences between them.

![Figure 1. Heat bar from Mobius SLIP](image)

There are more known rating-based peer-assessment systems than ranking-based systems. CrowdGrader (De Alfaro and Shavlovsky 2014) is a system that allows reviewers to give numerical scores to the artifacts they review. Based on the configuration, the numerical score can either be a holistic score if no detailed rubric is used, or scores based on detailed rubrics. Expertiza (Gehringer 2009) also supports both holistic and detailed review rubrics, but it allows more flexibility when instructors tailor their own review rubrics. Expertiza supports
10 different question types including dropdowns, checkboxes, textual feedbacks, etc. Expertiza also allows instructors to organize their review rubrics in tables or sections.

Other than ranking and rating, other designs have been implemented in different systems to improve the peer-assessment quality. MechanicalTA (Wright, Thornton, and Leyton-Brown 2015) has a feature that divides student reviewers into two groups, the supervised group and independent group. All students start off in the supervised group, but the reliable reviewers can be promoted by teaching staff to the independent group, which means their reviews will be considered as reliable unless any author complains about their reviews. Once a complaint of an independent reviewer’s assessment has been upheld, the reviewer will be placed back into the supervised group, whose reviews will be checked more frequently by the teaching staff.

Those different functionalities and design emphases of these peer-assessment systems lead to different schemas used for their data, which complicates the work of comparing different designs. For example, there are many dimensions on which we can compare ranking-based and rating-based systems such as rating accuracy, or usefulness of formative feedback. However, the comparisons can be challenging because 1) no system provides both ranking and rating as choices for instructors to use in their assignments, 2) researchers need to learn the design and terminology of each system before analyzing the data.

In this chapter, we introduce a Peer-Review Markup Language to provide a common definition of terminology across multiple systems. We used this markup language to facilitate data sharing between different systems. We discuss issues raised during this process and our approach to solving them.

2.2 Related Work

Many domain experts start with defining a set of terms first before data sharing (Noy, McGuinness, and others 2001). The common terms can help to annotate and share information in a large-scale and standardized structure. For some fields, this work has been
done very early, e.g. SNOMED clinical terms in medicine (Stearns et al. 2001) and FuGE in functional genomics (Jones et al. 2007). Domain experts want to create a common terminology because it can help researchers in several different ways:

To make sure same concepts are referred when same terms are used. Having a common set of terms is the basis of data/knowledge sharing and reuse (Musen 1992). However, in fact, several research projects on educational peer assessment use the different terms for the same concept, or use same terms for different concepts. For example, some research papers use “items” for students submissions (Diez Peláez et al. 2013), some others use the term “artifacts” (Song, Pramudianto, and Gehringer 2016). Arguably, an instructor may require students to submit multiple items (e.g., files, hyperlinks, videos, etc.) but they can be considered as a single piece of student’s work for one assignment, namely, one artifact.

To make the data sharing/combination possible. There are common attributes needed by different designs, for example, the notion of time. However, different database system/programming language may support different notions of time, e.g. as intervals, or using relative measures, etc (Noy, McGuinness, and others 2001). If the standard type of each notion is also defined by a group of domain experts, less data cleaning is needed when data is shared between researchers. Similar work has been done in systems biology community in 2003, for which Hucka et al. defined a Systems Biology Markup Language to provide a format for model sharing (Hucka et al. 2003).

To separate domain knowledge from operational knowledge. “We can describe a task of configuring a product from its components according to a required specification and implement a program that does this configuration independent of the products and components themselves” (Noy, McGuinness, and others 2001; McGuinness and Wright 1998). Researchers, especially in collaboration, may be interested less interested in implementation details. Therefore, separating the conceptual facts (domain knowledge) that researchers are interested in from implementation-related facts (operational knowledge) is also one of the goals of this project. For example, some peer-assessment systems are
designed to be assignment-based (like Expertiza (E. Gehringer 2009), CrowdGrader (De Alfaro and Shavlovsky 2014)): in one assignment, an instructor can specify multiple deadlines for students to select topics, form teams, submit their artifacts, and peer-review each other. Some other systems are task-based (like Critviz (Tinapple, Olson, and Sadauskas 2013)). This design does not have the notion of assignment, but a series of tasks (and, making it even harder to understand, in Critviz, each task is called an “assignment”). Though the design of task-based and assignment-based systems are different, the general flows of peer assessment are almost the same, therefore, whether the peer-assessment systems is task-based or assignment-based can be considered as operational knowledge. This kind of operational knowledge should not be researchers’ focus unless they are interested the implementation of those systems.

Unfortunately, the educational peer-assessment research community still lacks accepted standards for system components and workflow. To create domain terminology and their relations (and further, to create domain ontology), there are common steps suggested by Noy and McGuinness (Noy, McGuinness, and others 2001) (the process of defining PRML is also added for each step):

**Step 1. Determine the domain and scope.** The domain terminology is for the educational peer-assessment research field. It can also incorporate academic peer assessment (e.g. peer review of research publications).

**Step 2. Consider reusing existing ontologies.** We considered some existing ontologies/DB models like course content ontology (Boyce and Pahl 2007), Learning Tools Interoperability (LTI) (IML Global Learning Consortium 2017), E-Learning Educational Systems ontology (Gascueña, Fernandez-Caballero, and Gonzalez 2006), and MOOCDB (Veeramachaneni et al. 2013; “MOOCdb” 2017). Some terms from these can be reused but significant modification/extension is still needed (mainly because they are not designed for educational peer assessment). Table I shows some existing domain terminologies and relations defined and the reasons they cannot be reused in this project.
<table>
<thead>
<tr>
<th>Domain related</th>
<th>Num. of concepts</th>
<th>Why cannot be reused</th>
</tr>
</thead>
<tbody>
<tr>
<td>LTI</td>
<td>12</td>
<td>Lacking fine-grained entities</td>
</tr>
<tr>
<td>MOOCDB</td>
<td>20</td>
<td>Highly hierarchical</td>
</tr>
<tr>
<td>E-learning ontology</td>
<td>16</td>
<td>Lacking assessment-related definitions</td>
</tr>
</tbody>
</table>

Step 3. **Enumerate important terms.** This step was performed by inviting the designers/developers of four educational peer assessment systems to add the key terms to an online spreadsheet.

Step 4. **Define the properties of classes.** The designers/developers of 4 educational peer assessment systems were invited to fill in an online spreadsheet jointly to group the common terms used. The classes were identified considering all the database schemas and terms used in all the systems.

Step 5. **Define the attributes of the classes.** The designers/developers listed all the attributes of each class in their own products, then the operation-related attributes were removed (only domain-related field were kept). Personal information of users was removed as well.

Step 6. **Create instances.** Done by ETL (Extract, Transform and Load) projects. Find more details in Chapter 3.

### 2.3 Relational Peer Review Markup Language

Our Peer-Review Markup Language (PRML) was designed to be a generic data model/schema for modeling most of the data – both numerical and textual – generated by online peer-assessment systems used by students. PRML was originally designed jointly by the owners/designers of four online educational peer assessment system: Expertiza (E. Gehringer et al. 2007; E. Gehringer 2009), Mobius SLIP (McLean 2013), Crowdgrader (De...
Alfaro and Shavlovsky 2014), and CritViz (Tinapple, Olson, and Sadauskas 2013). All four systems are in active use and development, with new users, new functionalities, and new patches.

The first version of PRML is a relational version. It defines entities in educational peer assessments and the relationship between them, as shown in Figure 2. Peer assessment is usually performed within a Course, which may have one or several Assignments. The Course can have many Participants enrolled. Those individual participants may need to form teams or act alone in some cases; e.g., some systems support Students working in a team and submit an Artifact. Therefore PRML uses the concept of Actor to model an individual participant or team that undertakes any peer assessment activity. An Actor can either be a reviewer or a reviewee. An Actor may consist either of one participant, or multiple participants acting as a team. This design allows PRML to model the data generated by an individual reviews another individual’s work, an individual reviews a team’s work, a team reviews an individual’s work, or a team’s review another team’s work. Actors (individuals or teams) are also the authors of Artifacts. Those artifacts may relate to a common task (e.g., all students are doing the same homework assignment) or different tasks (e.g., each student or team is working on a different topic).

To facilitate peer assessment, instructors may also create their own review Rubrics. A Rubric can either be holistic – contains only one Criterion asking the reviewers to give overall comment and score an overall artifact – or detailed – contains multiple Criteria and scored on different aspects. Each Criterion is a prompt asking reviewers to evaluate some aspect/dimension of the artifact. Criteria may be of different types, such as textbox questions, rating questions, checkboxes, or ranking questions. The response that a reviewer filled in for the criterion is called a Critique (as REV_CRIT_LEVEL in relational PRML). A Critique can be textual or numerical or even both (to accommodate systems that ask for both numerical and textual feedback in one prompt).

In the design phase of PRML, we tried to make the PRML as generic as possible so that we could model the data generated by different systems. For example, we are aware of both
systems designed to support individuals’ work and systems designed to support teamwork. There are even systems can handle both. Therefore we introduced the concept to Actor to support both. In addition, we also considered and made sure that the PRML can support both holistic rubrics and detailed rubrics.

In addition, some data related to class settings are important for researchers. Those data are not necessarily represented in the database of the peer assessment system – they could be

Figure 2. PRML main concepts and their relations
some rule enforced by the system, or set up by the instructor. PRML is also designed to collect those data. Below are some examples:

**Reviewer assignment strategy.** Most systems start the peer-review process after the submission is due. Reviewers may be assigned either statically or dynamically. If they are assigned statically, either the instructor or some algorithm decides who will review whom. However, the dynamic assignment is often preferred because inevitably, some students will fail to submit their reviews, and thus, some submissions will be short of feedback. Dynamic assignment assigns submitted artifacts to reviewers on the fly. This allows a reviewer to be assigned, for example, to the submission that has received the fewest reviews so far. In reality, many more factors are at play when assigning reviewers. For example, if students are working on different topics, the system may allow reviewers to choose the topics they want to review. Another strategy is to use reviewer’s reputation (determined, for example, by how closely a reviewer’s ratings matched that of the teaching staff on a previous assignment) (Song, Hu, and Gehringer 2015b) to make sure that each artifact is reviewed by at least one competent reviewer.

**Degree of anonymity.** Most systems use some degree of anonymity in the review process. Anonymity may be applied both to the reviewers and authors (double blind) or only to one side (single blind). Some systems adopt anonymity in the review process, but then publish the artifacts and reviews to provide a sense of accountability for the authors and reviewers.

**Rejoinder/back-review policy.** Some systems allow authors to give back-reviews, or “rejoinders” to the reviewers. This feedback can be qualitative (prose, checkboxes) or quantitative (e.g., Likert scale), depending on the rubric. The rejoinder may or may not be included in the computation of the reviewer’s grade.

### 2.4 Star Schema of Peer Review Markup Language

Based on the entities and relations defined in relational PRML, we have also created a star schema (Kimball and Ross 2002) of PRML to facilitate data sharing across different online
educational peer-assessment systems. This data-warehouse schema was designed based on the star schema, which separates the data into facts that hold measurable, quantitative data about the peer assessment and dimensions, which are descriptive attributes related to the fact data.

In our data-warehouse schema (Figure 3), the fact table in the middle is the Critique table. It contains information from textual and/or quantitative feedback based on a single criterion. This is also the finest grained data which can be created in educational peer assessment process. Around the Critique table, there are many dimension tables which include information on additional attributes of this Critique. The Actor table contains references to both reviewer and reviewee for each critique. The actors can either be individuals or a group. The Participant table stores information on users who participate in an assignment. Between the Actor table and the Participant table, there is another joint table called Actor_participant, which is designed to maintain the $n$-to-$n$ relationship. However, since this data warehouse is designed to be anonymized, we barely need to store any data in Participant table. The Eval_Mode records the evaluation mode of this critique, which can be ranking-based, or rating-based, or even both.
The Criterion table holds information on the criteria in the review rubrics, including scale, weight, and type (textbox, rating, checkbox, etc.). A Criterion can either be holistic or detailed. The Task table contains information on when the review period starts and ends, the CIP (Classification of Instructional Programs) code of the course, the type of the task (submission, review, etc.) and further information for the assignment and course. Note that we do not use a separate assignment or course table, instead, we "flatten" the schema and put all the information related to assignments and courses into Task table. This is to reduce the number of join-table queries and thereby speed up querying of the data warehouse. The Artifact table contains information about the student’s work for each task. Considering that the artifact may contain many items (e.g., a .pdf file for project report plus a video for demonstration), we also created an Item table. The Item table is also associated with the Critique table, since reviewers
may also create some items (e.g., a reviewer may be able to add comments to the original .pdf file and submit it to the online peer assessment system).

In addition to the information above, the data warehouse also collects the meta-information about the assignment settings in the Course_setting table. The information includes the anonymity setting (none, single blind, double blind, etc.), the workflow (in term of tasks or assignments) and the rubric mode (holistic or detailed). This table facilitates further comparisons on different assignment settings after enough data has been collected.

In the star data-warehouse schema, all the entities except for Actor_Participant and Participant are directly associated with the fact table Critique. This can make the queries easy to implement and fast enough even when the data volume is huge. Any comparison based on this data warehouse can be considered as "slicing" the data using one or more dimensions as constraints. For example, we can create a query to retrieve all the reviews generated by ranking-based assessments and another query to get all the reviews generated by rating-based assessments. Then, a comparison can be done using metrics such as the average length of comments or the number of suggestions offered. In this way, we can come up with a more general conclusion about which type of assessment is better.

2.5 Evaluation

In this section, we explain how PRML models common concepts modeled in peer-assessment systems. We mainly focus on the relational version which is also the original version. Since this PRML is a conceptual model for the common entities in peer assessment, we compare the entities we defined in PRML with the concepts that used in different peer assessment systems.

To be a common data model for peer-assessment, PRML should meet both standards below:

- The data modeled are the common data in peer-assessment.
The data not modeled are uncommon or unnecessary.

Section 2.5.1 examines if the entities defined in PRML are also supported in actual peer-assessment systems. Ideally, those entities in PRML should have corresponding concepts in those systems as well. Section 2.5.2 examines the concepts in those peer-assessment systems that do not have corresponding entities in PRML. It is acceptable that some concepts in some peer-assessment systems are not modeled in the PRML, but there should be reasonable explanations.

In this section, we only compare the concepts in the PRML and the actual peer-assessment system. We only compared the entities and skipped all the relationships (e.g., there is a 1-to-\( n \)) relationship between assignments and courses). The verification of relationships will be described in Chapter 3.

2.5.1 Verification of Concepts in PRML

We verify the original PRML by examining whether the concepts (terminologies) identified have corresponding ones in the design of some known peer-assessment systems. Table II is a spreadsheet showing whether the concepts of PRML exist in the actual design of the peer-assessment systems we collaborated with. The first column of Table II are the models we identified in the PRML. If a peer-assessment system has the same concept, even by a different name, we recorded a “✓”, otherwise we recorded a blank. Please note that a peer-assessment system does not need to have that concept as a separate table in the database to count as “✓”.

...
Table II. Whether there are corresponding concepts/terms in actual peer-assessment systems

<table>
<thead>
<tr>
<th>Concept</th>
<th>CritViz</th>
<th>CrowdGrader</th>
<th>Expertiza</th>
<th>Mobius SLIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>Artifact</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Assignment</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>CIP level</td>
<td></td>
<td>✔</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Content case</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Course</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Criterion</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td></td>
</tr>
<tr>
<td>Critique</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td>✔</td>
</tr>
<tr>
<td>Item</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Level</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Organization</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>✔</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Participant</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peer-review app.</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task type</td>
<td>✔</td>
<td>✔</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From Table II we find that most of the concepts we identified in PRML have corresponding concepts/terms in the peer-assessment systems we investigated. There are two PRML concepts that are not supported: CIP level and Peer-review application. The purpose of the Classification of Instructional Programs (CIP) is to provide a taxonomic scheme that will support the accurate tracking, assessment, and reporting of fields of study (“Classification of Instructional Programs” 2017). We argue that CIP level is necessary since one of the purposes of this project is to facilitate data sharing on peer assessment data from a variety of institutions and subjects. Those subjects may be named differently depending on the institutions and instructors. In this case, CIP levels provide common identifiers for the same subjects. The Peer-review application is a concept only useful for shared data when data from different systems have been combined. Therefore, it is not surprising that none of the investigated systems has this concept in their designs. But in the ETL process, the peer-assessment system manager should be in charge of filling this information before sharing the data to the data warehouse.
A “✓” in parentheses in Table II indicates that a peer-assessment system partially has the concept defined in PRML. CrowdGrader and Expertiza partially have the concept of Actors, for example. The reason is that in the ideal PRML design, both reviewers and reviewees are considered to be actors. In CrowdGrader and Expertiza, however, only reviewees are handled as actors. Mobius SLIP partially supports the concept of Criterion because it only supports holistic review rubrics, but not detailed rubrics. In other words, Mobius SLIP does not allow instructors to customize their own detailed rubrics to guide students in peer-assessment activities.

2.5.2 Verification of Completeness

Table II only shows that most of the concepts defined in PRML have correspondent concepts in actual peer-assessment systems. Another part of the verification is to show that the concepts/terms that used in actual peer-assessment systems which are not correspondent to any PRML concepts are not necessary for data collection for research use, or not common. Table III lists some concepts/terms that are not related to any PRML terms in each peer-assessment system.

From Table III we found that the system investigated may have different other concepts built into those systems. This is not surprising to us since those systems are designed to provide peer-assessment functionality according to the designers teaching styles of the requirements of the courses originally supported. CritViz and CrowdGrader are two systems which support relatively fewer additional concepts. CrowdGrader supports uses the concept of “SimiCheck setting” since it uses the SimiCheck web service to check the similarity between students’ submitted artifacts (section 3.4.3. will cover this in more detail). This is not a concept we would bring to PRML since this is special to CrowdGrader.
Table III. Additional concepts/terms in peer-assessment systems

<table>
<thead>
<tr>
<th></th>
<th>CritViz</th>
<th>CrowdGrader</th>
<th>Expertiza</th>
<th>Mobius SLIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Review group</td>
<td>SimiCheck setting</td>
<td>Bid</td>
<td>Review group</td>
<td>Discussion board post</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bookmark</td>
<td></td>
<td>Discussion board vote</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Penalty</td>
<td></td>
<td>Course-enrollment request</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Late policy</td>
<td></td>
<td>Coupon</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Join-team request</td>
<td>Course student billing</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Invitation</td>
<td></td>
<td>Delayed job</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Delayed job</td>
<td></td>
<td>Delayed job</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Topic suggestion</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Both CrowdGrader and Mobius SLIP have the concept of “Review groups”. This concept is related to the peer-assessment setups in those two systems. Both systems organize students into review groups first, then the students in each team review each other’s artifacts. This setting itself is not brought into PRML, but we have modeled this information in a different approach: we added a bool variable in COURSE_SETTING in star schema of PRML.

Mobius SLIP has certain concepts built into the system because it is a commercialized system. The concepts “coupons”, “course student billings” are related to the transactions and payments, which should not be modeled in PRML since they are personal information and not related to peer assessment. In addition, Mobius SLIP also has a discussion board built in, which is related to the concepts “discussion board votes” and “discussion board posts”. Since Mobius SLIP is the only system which has a built-in discussion board, and the discussion on it is not closely related to students’ peer-assessment behavior, we do not include those models into PRML.

Expertiza system also has some unique concepts supported, which are not closely related to students’ peer-assessment behavior. The concept of “bid” is related to the feature that allows students to bid for topics. The concept of “bookmarks” is related to students recommending resources that can be used in carrying out a project. The “late policy” and “penalty” are related to calculating penalties for late submitters or late reviewers. The concepts of “join team
request” and “invitation” are related to the features which allow students to invite others to form teams or request to join others’ teams. The concept “topic suggestion” is related to the feature that allows students to suggest topics for an assignment. Those features, however, are not common and not directly related to students’ peer-assessment behaviors.

2.6 Contribution and Conclusion

In this section, we have presented our work on a Peer-Review Markup Language (PRML), which defines the common entities and relations in the educational peer-assessment process. This common terminology makes it easy for researchers to discuss the process, and more importantly, share peer-assessment data.

Based on the original relational PRML, we further designed a star schema of PRML which makes data sharing across different systems much easier. Our project is currently comprised of four systems, of which two have already shared their data with us. The courses whose data is in the data warehouse cover a wide range of majors such as computer science, mechanical engineering, education, and art.

In the evaluation section, we verified the PRML as a conceptual model. We compared the entities defined in PRML with the actual designs of four peer-assessment systems. The comparison shows that 1) most of the entities defined in PRML have corresponding concepts in the designs of those peer-assessment systems; 2) for the concepts in those systems which do not have corresponding entities in PRML, there are good reasons. Those reasons could be 1) they are related to private data, 2) they are related to uncommon features in peer-assessment systems, or 3) the information is included in PRML, but with different designs.
CHAPTER 3 Data Warehouse for Educational Peer Assessment

3.1 Introduction

PRML defines a common terminology for concepts that are pervasively used in educational peer assessment. This common terminology allows researchers, system designers and owners to discuss common concepts related to their research and their systems. In addition, PRML also defines a generic schema to model the relationship of data generated by peer-assessment activities. Thus, PRML can help build a data-sharing protocol and thereby facilitate data sharing between educational peer assessment systems.

To collect data from different peer assessment systems, a data warehouse was built based on star schema of PRML. The data on users and reviews, plus the variety of different peer-assessment settings can help researchers to study and test hypotheses and come up with more general conclusions.

3.2 Related Work

Most researchers should agree that “data drives research/discovery” (Ricanek and Tesafaye 2006). Though research is more data intensive in recent decades than in the past, researchers still perceive barriers to data sharing. Tenopir et al. did a survey on a total of 1,329 scientists about their preservation/sharing experience. They found that insufficient time and lack of funding are main reasons that researchers have not made their data electronically available (Tenopir et al. 2011). Campbell and Bendavid also reported that amount of effort involved in accessing and sharing data is a top reason for researchers to withhold the data (Campbell and Bendavid 2002). This seems to be true of the educational peer assessment research community.

Tenopir et al. also found that researchers are somewhat satisfied with their short-term data lifecycle, but still struggle with long-term data preservation (Tenopir et al. 2011). This means that the data that researchers spend time collecting, cataloging, analyzing is likely to be lost.
after short-term storage. This is true for the educational peer assessment research community as well. Without a common schema, researchers may collect the data in the schema of the best convenience for their research project, but the data tend to be useless unless there is a continuing research after the original publication.

To preserve data and facilitate later research, cyberinfrastructure will play a major role in the future in data sharing for research purpose (Tenopir et al. 2011). There are several online data shops which facilitate data storage and sharing, e.g. PSLC Data Shop (“PSLC DataShop” 2017), HEDS Consortium (“HEDS Consortium - Home,” n.d.) and Dryad Digital Repository (“Dryad Digital Repository,” n.d.). Those data shops facilitate data storage with very low or no cost.

One potential problem with sharing data in the PSLC Data Shop is that data access is an extra hurdle for researchers: for the owner of data, they have to upload patches or new versions of their data; for the data consumers, they have to download the data by themselves. This is especially inconvenient for research projects which are based on data from different resources. The researchers may have to download many zipped data files and attempt to combine and clean the data to create a dataset. Considering that the data in those data shops is not necessarily in the same schema, combining this data may require tremendous time.

Another approach to sharing data is to host a data warehouse with read-only access open to the public and write access to data contributors. This approach makes the data access easier for researchers – they do not need to download the packages of data and worry about whether the data is up-to-date or combinable. Instead, they only need to learn one data schema to access the open data warehouse. This also facilitates data sharing since the data shared is in the same schema – the researchers who are willing to share their data can request write access and contribute the data into data warehouse. The software development and data sharing, in this case, are not necessarily performed by the same laboratory, and do not necessarily benefit one single laboratory either, which solves the dilemma observed by Paton (Paton 2008). For example, researcher A has created a tool to generate a global ranking of the entire class based on peer assessment results and visualize it. If another researcher, B, wants to create another
tool to calculate the peer assessment variance of each artifact, (s)he can reuse part of A’s project since both projects fetch the same data from the same source.

A lot of data from MOOCs has also been used in some recent research projects (Kulkarni et al. 2013; Piech et al. 2013). The data from MOOC platforms can easily encompass peer assessment data from thousands of students, e.g., an HCI course on Coursera had more than 5000 students (Kulkarni et al. 2013), which is more than most of the course sizes on other peer-assessment systems. So why would we still try to share data between different peer-assessment systems instead of using the data from MOOCs? The most important reason is that the data shared by different systems exhibits greater variety than the data from MOOCs.

The variety of peer-assessment design is one of the research topics of great interest. Researcher have been investigating many possible approaches to improving peer assessment (E. F. Gehringer, Ma, and Duong 2016), and a variety of attempts have been made, such as:

- evaluating the quality of the peer-assessments and making the results available to students (Patchan, Schunn, and Clark 2017; Yadav and Gehringer 2016);
- using calibration as a training before students enter they enter the peer-assessment phase to review others’ work (Balfour 2013; Russell 2001; Song et al. 2016);
- building measures of review accountability by allowing authors to “quiz” their reviewers (Song, Hu, and Gehringer 2016);
- making reviewers’ reputations visible to authors and reviewers (O’Toole 2013; Song, Hu, and Gehringer 2015b; Hu, Song, and Gehringer 2016);
- allowing multiple rounds of review so that authors can improve their artifacts using the assessment they received (Song et al. 2016);
- making the peer assessment anonymous (Theising, Wu, and Sheehan 2014),
- anonymous, or anonymous first then onymous (Tinapple, Olson, and Sadauskas 2013).

Those different techniques have been applied in various online peer-assessment systems, but rarely supported on MOOC platforms. On MOOC platforms, the peer-assessment strategy
is usually anonymous, pre-assigned and only for one round, which is the most basic setting in most of the known peer-assessment systems.

Another aspect of the peer-assessment data from MOOC platforms is that MOOC students are atypical of student populations. The percentage of MOOC students who completed the enrolled course is usually low, e.g., Pursel et al. reported a 5.6% course completion rate among enrolled students. In addition, students who are able to finish courses in MOOC platforms are comfortable with self-directed learning and more intrinsically motivated (Barak, Watt, and Haick 2016). Wang and Baker’s research on MOOC students also supported this: they found that the completers tend to be more interested in the course content, yet the non-completers tend to be more interested in joining MOOCs as an experience (Wang and Baker 2015). From this perspective, the MOOC students are different from traditional college students: the successful MOOC students are as good as, if not better than traditional students, but the unsuccessful MOOC students can be much weaker.

In summary, we found that it is essential to improve data sharing for the educational peer-assessment research community. Without this data sharing mechanism, the source of research data would either be data from MOOC platforms, or researchers’ local data (most likely generated in the courses they taught). The researchers’ local data is usually not voluminous enough to derive any general conclusions. Due to the difference between MOOC students and college students, findings based on MOOC data may not be helpful in traditional course settings.

3.3 Data Sharing between Peer Assessment Systems and Data Warehouse

PRML defines common terminology and common schema for sharing data, which saves time for data sharing – researchers do not need to understand different data schema from different systems, instead, they only need to understand PRML and know how to transform their data into PRML schema (if they have a system).
3.3.1 Schema Used

The data warehouse uses the star schema of PRML to store the shared data. There are several reasons for using the star schema (presented in Section 2.4) instead of the relational schema of PRML.

First, the star schema reduces the complexity of join table queries, which makes the (complex) queries faster (Chaudhuri and Dayal 1997). The data contributed by different peer assessment systems is more used for summarization and consolidation, instead of transactional purposes (it is unlikely that researchers are interested in a single review or a single reviewer's record). The star schema of PRML is multidimensional—each dimension table represents one or some dimensions of interest. Typically, the researchers are more interested in the results of rollup operations, e.g., the score distribution of all the reviewers in one assignment. If this query is done on relational PRML, almost 10 tables need to be joined in the same example, which may result in unacceptable delays. By contrast, this operation can be done much faster with a star schema than using a relational schema (only two dimension tables need to be joined with the fact table).

Second, the star schema also makes data sharing easier for researchers who would like to share their data. If the developers/owners of educational peer assessment systems need to transform data into relational PRML, the process may take much longer because there are more entities in relational PRML and some of them may not have corresponding ones in the design of some peer assessment systems. In this case, if the data needs to be converted from the original schema of peer assessment systems into relational PRML, new mock entities have to be created. Creating those mock entities can be even harder when the relation between entity associations in the peer review system design are not the same as the associations in relational PRML design.

For example, Figure 4 is part of the design of relational PRML (unrelated attributes removed) and Figure 5 shows the corresponding part of design in Expertiza. To make the comparison easier, terms in PRML is used in this example, though different terms may be used in the real Expertiza design.
In the Expertiza design, as shown in Figure 5, the Actors are associated with Assignments. However, in relational PRML, the Actors are associated with Tasks. Since there is no corresponding Actor_Task table in Expertiza, records for Actor_Task have to be created. Namely, developers must create Actor_Task entries for all tasks, and all actors. To transform the data into relational PRML, a large amount of similar work must be done, which means an extra burden for data sharing. In addition, more often than not, these kinds of tasks have to be done by developing a data-transforming application.

With the star schema of PRML used in the data warehouse, the Assignment and Task tables in the original Expertiza design will be combined into the Task dimension. The $n$-to-$n$ Actor-Participant mapping is still the same. The Actor_Task table becomes part of the Critique table. This transformation can be done much faster because the designer can use join-table queries to "flatten" their data instead of transforming their data from one relational schema into another relational schema. This kind of task, if the database schema follows normal forms, is likely to be handled by queries instead of coding.
3.3.2 General Design

The data sharing between an online peer-assessment system and the data warehouse can be considered to be an Extract, Transform and Load (ETL) process (Vassiliadis 2015). Figure 6 shows the process for the data-sharing process for two systems, Expertiza and CritViz. The data is extracted from the source databases, then transformed into the data-warehouse schema and loaded to the staging data warehouse first. (The staging data warehouses serve as temporary repositories for data-verification purposes.) This process is done using Pentaho (Pulvirenti and Roldán 2011). The central data warehouse can pull the data from the staging data warehouses after the data is validated.
Besides transforming the data into the data-warehouse schema, the ETL process also anonymizes the original data by skipping all the personal information and not including it in the staging data warehouse. The central data warehouse, thereby, will exclude any personal information of the users. The data warehouse is purely for research uses and need not include any personal data such as names, emails or campus IDs.

### 3.3.3 Usage of ETL Tool

The ETL processes from the original databases to the staging data warehouses differ for each system. Depending on the design of the peer-assessment system and its own database schema, some ETL tasks can be straightforward: developers just need to use join table queries to flatten the data to transform the data into the star schema. Figure 7 shows an example of this ETL process. There are five entities from a relational schema of Expertiza system on the left side. Those entities will be flattened into the TASKs table in the star schema. The ETL developer only needs to create a join-table query for those five tables and to extract the useful information, then map them to corresponding fields in the star schema of PRML on Pentaho.

![Figure 7. An example of Extract, Transform and Load (ETL) process](image)

Figure 8 is a screenshot of Pentaho. The main panel of this user interface shows the join table query for combining the necessary information in Figure 7.
However, coding is still necessary in some cases. ETL developers may need to split original data from one table into several tables, or add new data or relationships in this ETL process. For example, in the data-warehouse schema, both reviewers and reviewees are actors, which can either be individuals or teams. The actors are the owners of artifacts for each assignment. This design is to make the data-warehouse schema more generic. Figure 9 shows this part of the star schema of PRML with unrelated fields removed.
However, this is not the case for all the systems. For instance, the design in Expertiza only allows reviewers to be individuals, which is Participants in star schema of PRML. Therefore, the relationship between reviewers and reviewees is Actor-Participant mapping instead of Actor-Actor mapping in PRML (to make the discussion easier, the entities are all named the same as in PRML, the unrelated attributes are removed as well), which is shown in Figure 10.

In the ETL process, therefore, new Actor entries need to be created for individual reviewers as Actors. This process can be carried out by ETL tools as well, however, coding is necessary to create new entries in Actors table.
3.3.4 Designing ETL Applications

In some cases, the peer assessment system is not designed based on a relational database, or part of the data is not even stored in a relational database, which makes the ETL process even harder. In this kind of cases, the ETL developers may need to create their own ETL applications instead of using ETL tools like Pentaho. For example, one of the systems which promised to share data, CrowdGrader, uses key-value tables to store part of the data. Therefore, queries on Pentaho do not work due to the potential flexibility of key-value tables. In this case, the ETL developers must collaborate with the star-schema designers to implement an ETL application for CrowdGrader. The design phase has been finished. The detailed project design documentation is open to the public. ¹

3.4 Evaluation

In this section, we verify both versions of PRML as a data sharing protocol. As discussed above, the relational PRML is a conceptual model and the star schema of PRML is more suitable for data sharing. Section 3.4.1 discusses the suitability of relational PRML as a data-sharing protocol, Section 3.4.2 discusses the suitability of star schema of PRML for the same task. By comparing the estimated workload, we show that the star schema is a better choice considering the workload and complexity. In Section 3.4.3, we use the web services developed in PeerLogic project as examples to illustrate the power of the star schema as a data protocol for designing web services.

3.4.1 Evaluation of ETL Process Based on Relational Schema

In relational PRML, we identified 18 different concepts and 24 relations between those concepts (presented in Section 2.3). If relational PRML is used as data-sharing protocol, all those entities will be transferred into data records and relations need to be kept as well. For example, if a student with id 1122 has been enrolled in a course with id 34, there should be a Participant record with id 1122 to model the student, and a Course record with id 34 to model

¹ http://bit.ly/2ujTR8u
the course (please note that, this is an example for discussion, not exactly the design for the ETL process). To model the fact that the student has been enrolled in the course, there should also be an Enrollment record indicating student with id 1122 has been enrolled in course with id 34. Figure 11 shows the expected data after proper ETL process.

From this example, we can identify two main parts of the ETL process for data sharing: create the models and create corresponding relations. In the easiest case – the client peer-assessment system has corresponding concepts as models, and there is the same relationship between those models – the ETL is straightforward. For the example of the n-to-n relationship between Participants and Courses defined in relational PRML, Mobius SLIP has corresponding models (called courses and students) and a corresponding relation (called courses_students). The ETL developer can extract the relevant information from Mobius SLIP without additional work.

From Figure 12 we find that though same information may have different names in Mubius SLIP data schema and relational PRML, the mapping is still clear since the n-to-n relation is the same in both schemas.
In some other scenarios, however, the ETL process can be more than transforming existing models and relations from the client systems into PRML format. For example, the $n$-to-$n$ relationship between Participants and Courses defined in relational PRML does not work for CrowdGrader. In CrowdGrader, there is no concept of Course as defined in PRML. Instead, there are assignments, and all the students are enrolled in assignments instead of courses. The ETL process in CrowdGrader, therefore, is different from the case of Mobius SLIP: the ETL project needs to create course records first, then transform the $n$-to-$n$ relation between assignments and students to $n$-to-$n$ relation between courses (after being created) and students. This means the same task, in CrowdGrader, will take more time, and has to be done with the collaboration of the system admin(s) or instructors.

Figure 13 illustrates the ETL process of the $n$-to-$n$ relation between Course and Participants from CrowdGrader to relational PRML. The Participant and Assignment defined in PRML have corresponding records in the CrowdGrader schema. However, there is no concept of
Course in CrowdGrader. Therefore, the Course records in the relational PRML have to be created in the ETL process first, then the Enrollment records also need to be created. In this case, the difference of design between CrowdGrader and relational PRML make the ETL process complicated.

Figure 13. ETL process of $n$-to-$n$ relation between COURSE and PARTICIPANT from CrowdGrader to relational PRML

To evaluate how much of the ETL project can be done by simply transforming existing models and relations (as the case of Mobius SLIP), and how much cannot be done so easily, we created Table IV. The first column records all the entities defined in PRML as first 1/3, the $n$-to-$n$ relations as the second 1/3, and the 1-to-$n$ relations as the last 1/3. Then for each studied peer-assessment system, we record whether there are any entities or relations that can be reused in the ETL process. The first 1/3 of Table IV is the same as Table II.

For the rest of Table IV, we recorded “$\checkmark$” if:

- The client system has corresponding entities in the design, and
- There is $n$-to-$n$, 1-to-$n$ or one-to-one relationship between those entities.

We recorded “($(\checkmark)$)” if the relations in the relational PRML is partially same, or same in some cases to the design in client systems. Otherwise we record a blank.
Table IV. Whether existing models/relations can be reused when ETLed into relational PRML

<table>
<thead>
<tr>
<th></th>
<th>CritViz</th>
<th>CrowdGrader</th>
<th>Expertiza</th>
<th>Mobius SLIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>No</td>
</tr>
<tr>
<td>Artifact</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Assignments</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>CIP level</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Content case</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Course</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Criterion</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Critique</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Evaluation mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Item</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Level</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Organization</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Participant</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Peer-review app.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Task</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Task type</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Actor - Task</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actor-Participant</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Artifact - Critique</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Course-Owner</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Criterion - Level</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Organization-Owner</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Organization-Participant</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Participant - Course</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Peer review app.-Owner</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task - Criterion</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Task - Content case</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Artifact - Actor task</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Artifact - Item</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Content case - CIP level</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Course - Assignment</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Course - CIP level</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Course - Organization</td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>Course - Peer review app.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Critique - Eval. Mode</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Critique - Level</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
From Table IV we found that, in the four systems we investigated, a total of 168 models and relations need to be handled in the process of sharing the data from those four systems. Out of those 168 items to ETL, 88 of them are “(✓)” or blank. This means that among 168 potential ETL tasks, only less than half of them can be transformed directly from existing design.

3.4.2 Evaluation of ETL Process Based on Star Schema

In this section, we provide our argument for the suitability of using star schema of PRML as the data sharing protocol between peer-assessment systems. We also compare the potential workload between using original PRML and using the star schema as data sharing protocol.

To evaluate what fraction of the ETL project can be done by reusing flattened models and their relations (also after flattening) such as the example in Figure 10, we created Table VI. The first column records all the entities defined in star schema of PRML in the first set of rows, and the relations are listed in the section below.

For the rest of Table VI, we recorded “✓”, “(✓)” or blank based on the following rules:

- “✓” if the client system, after being flattened properly, has the corresponding entity/relation.
- “(✓)” if the client system, after being flattened properly, has the corresponding entity/relation in some cases.
- blank otherwise.
Table VI. Whether existing models/relations can be reused when ETLed into star schema of PRML.

<table>
<thead>
<tr>
<th></th>
<th>CritViz</th>
<th>CrowdGrader</th>
<th>Expertiza</th>
<th>Mobius SLIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors</td>
<td></td>
<td>(√)</td>
<td>(√)</td>
<td></td>
</tr>
<tr>
<td>Actor_participants</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Artifacts</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Criteria</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>(√)</td>
</tr>
<tr>
<td>Critiques</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Items</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Participants</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Tasks</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Actors – Critiques</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Actors – Participants</td>
<td></td>
<td>(✓)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Artifacts – Critique</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Artifacts – Items</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Artifacts – Tasks</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Critiques – Criteria</td>
<td></td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Critiques – Items</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Critiques – Tasks</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From Table VI we found that, in the four systems we investigated, a total of 64 (flattened) entities and relations need to be handled in the ETL process. Among those 64 items to ETL, 31 of them are “(✓)” or blank. This means that more than 51% of the ETL project can be done by reusing models and relations after the data models are flattened. This is higher than the reusable (with checkmarks) percentage of relations and models if we use the relational PRML (48%). What is more important is, the number of items that cannot be reused decreased from 80 to 31. The main reason that the percentage of reusable items (relations or entities) increased is the fact that flattened entities have potentially more relations. Therefore, the relations between the flattened entities in the original peer-assessment system design have a higher chance of being reused in the ETL process.

We can reconsider the example of CrowdGrader presented in Section 3.4.1. In CrowdGrader, the Students are enrolled in Assignments instead of Courses (using the terms of PRML). Therefore, extra work has to be done (either with more complicated queries, or extra coding) to transfer the relation from Assignment-to-Students into Course-to-Students. With the star schema of PRML, the information of Courses and Assignments are all flattened into
Tasks. Tasks, after being flattened, also have the relations that once were on the Courses or Assignment. In this case, no matter whether the original design of a peer-assessment system is Assignment-to-Students or Course-to-Students, they will become Task-to-Students after being flattened.

For all the unchecked entities or relations in both Table IV and Table VI, the ETL processing should still be performed either by more complicated queries, or coding in a higher level language. However, the difficulty of that ETL task is likely to remain the same without becoming more complicated.

3.4.3 Evaluation of Using Subsets of Star Schema in Web Services

Another use for PRML is that it can facilitate data sharing between peer-assessment systems as clients and web services. In the PeerLogic project, we have developed several web services which provide additional features/functionalities to different peer-assessment systems. To make sure those web services can easily be incorporated into existing peer-assessment systems, a well-defined protocol is also needed to wrap the data. PRML can also facilitate this type of data sharing between clients and web service servers.

The difference between sharing databases (with the data warehouse) and web service calls is that, when sharing data with the data warehouse, a comprehensive set of data is necessary. In the data sharing between different peer-assessment systems and the data warehouse, we need to guarantee that the data is clean and complete, e.g. all the Critiques should belong to a Task. If any Critique is not associated with a Task record, or is associated with a non-existent Task, it will be considered as dirty/incomplete data. This kind of data will be removed from the staging data warehouse (discussed in Section 3.3.2).

When sharing data for the web-service calls, however, we should minimize the unrelated data sent. Since the web services do not collect data, the client should not send data to the web service servers unless the data are relevant. For example, if a web service only needs (even part of) the information of Critiques, the information related to the associated Task should not be sent. In other words, the schema used to share data between client systems and web services should be a minimal subset of the PRML and only include the necessary information.
Reputation web service. Reputation system is a common way to tell which peer reviewers’ scores are credible. This can be done by comparing scores assigned by different reviewers with each other. Figure 14 is an example from Hamer’s paper (Hamer, Ma, and Kwong 2005). The left side is review score matrix: there are four reviewers, \(a\), \(b\), \(c\), and \(d\) and four essays, 1-4. The numbers in the matrix are review scores. The reviewer \(d\) is supposed to be an unreliable reviewer since the reviewer scores assigned by \(d\) do not agree with others’. The table on the right side shows the reputations for those four reviewers.

Several reputation systems have been designed, but they are not convenient for the peer-assessment systems to use if they have to implement those algorithms locally first. To make the reputation algorithms pluggable for different peer-review system, we have created a reputation web service (Song, Hu, and Gehringer 2015b). In the reputation web service, the server accepts the total scores of each peer assessment instead of detailed scores on each rubric criterion. The Critiques transferred, however, can be considered as Critiques based on holistic review rubrics (Hu 2016). In addition, the clients do not need to share the textual information in Critiques.

Figure 15 is a screenshot of student authors’ view of the peer-assessment scores received from Expertiza system. The bottom half of this screenshot is the peer-assessment scores received in two review rounds. The Expertiza system displays the reputations of reviewers only for the second round (since the first round of this assignment is summative round). The range of reputation is from 1 (highest reputation) to 0 (lowest reputation). The student authors, from this view, can learn the reputations of their reviewers before reading more details about the feedback given by them.
Reputation web service was the first web service we developed based on PRML as protocol. I was the original designer of this web service (Song, Hu, and Gehringer 2015b). This work is later extended by Zhewei Hu (Hu 2016).

Summarization web service. In peer assessment, we have observed that students are less likely to read a large amount of feedback, especially when the number of reviews received is overwhelmingly large (Ferry Pramudianto, Chhabra, et al. 2016). Therefore, we created a summarization web service which automatically summarizes the feedback by grouping the similar content that is covered by different reviewers, which would capture the strength and weaknesses of the work.

In this web service, the server does not need any information about the numerical feedback, instead, the clients only need to send textual feedback. There are different levels at which peer-assessment systems may potentially want the textual content to be summarized, such as all the all the responses for one artifact, or all the responses for one artifact based on separate criteria, or all the responses for all the artifacts based on separate criteria. To facilitate the
different level of data for summarization, the server requests the finest granularity of data, which is Critiques.

Figure 16 is a screenshot of the summarized feedback to students authors. In this screenshot, the author has received 20 reviews, which is larger than the author is likely to read. In this case, Expertiza summarizes the feedback based on each rubric criterion and show the summarized feedback to the author instead.

![Score for Writing assignment 1b (Wikipedia)](image)

**Figure 16.** Display of summarized reviews for each criterion in Expertiza

In the summarization web service, I helped design the data sharing protocol for the web service calls, the main designer of this web service was Ferry Pramudianto (Ferry Pramudianto, Chhabra, et al. 2016).

**SimiCheck.** As part of the PeerLogic project, SimiCheck (“SimiCheck” 2017) is a web service that checks similarity among documents or source code. It can be used to analyze how similar the files being submitted are. After the comparison has been done, the web service returns a report on the level of similarities between all the students’ artifacts.
To call SimiCheck, a client peer-assessment system should prepare the submitted content for each assessee actors.

Figure 17 is a screenshot of plagiarism check report in Expertiza. This information is based on the returned information from SimiCheck. In this report, an instructor can find the similarities between students’ submissions in an assignment.

![Plagiarism Checker Report for Final Project (and Design Document)](image)

Figure 17. Display of plagiarism checker report for instructors in Expertiza

The SimiCheck web service was implemented by Dr. Luca de Alfaro in University of California, Santa Cruz. The Expertiza interface to SimiCheck was done by a final project team advised by me in 2017.

**Tile Chart.** Tile Chart visualization is used to map relationships between two entities where the intersecting point between the variables on $x$- and $y$-axes can be denoted by a specific numerical value along with informative text. The numerical value should belong to some range (specific minimum and maximum). The value will be denoted by a color mapped to that particular value so that it’s easier to compare different values belonging to different ranges. In this section, we use one use case which is to show an author all the received peer assessments based on a detailed rubric.

The information that needs to be sent to the web service server, in this case, is the textual and numerical feedback in Critiques in the client peer-assessment systems. In addition, the color scheme should also be sent to help the web service generate the visualization accordingly.

Figure 18 is a sample tile chart based on mock peer-assessment data. The $x$-axis is the submissions in an assignment and $y$-axis are the reviewers in this assignment. The color
scheme is set from green (very positive sentiment) to red (very negative sentiment) in advance. Each tile in this visualization represents a peer assessment. The user can also click on each tile and the tile will expand and show the text related to that review.

The tile chart web service is originally created by an independent student directed by Ferry Pramudianto. I advised this project in designing the data sharing protocol.

**Team Formation.** The team-formation web service helps students to form teams based on their project preferences. In the sign-up phase, students can submit their project preferences in order (in Expertiza which is a system currently using this web service, students can submit as many preferences as they want to, rank-ordering them). After this phase, the peer-assessment system will submit those data to the team formation web service, and the web service will try to cluster students into groups to maximize the students’ satisfaction by assigning them to the higher-preferred topics with the students.
When the client systems submit the students’ preference information, the most important data to transfer to the server is the students’ preferences (called bids in Expertiza). In addition, students may have formed teams before submitting their preferences as a team. Therefore, the information of Actors, Participants and Actor_Participants should also be sent to the server.

Figure 19 is a screenshot of how the students submit their bids for topics in Expertiza. All the topics were at the left side originally. Students can visit this page and pull their preferred topics to the left side. They can also use dragging to order the topics on the right side to make their first priority on the top. The color of each topic is based on the number of bids that other students bid this topic. The green color means there are a relatively smaller number of bidders and red means there are more bidders.

![Figure 19. Students' user interface for bidding feature in Expertiza](image-url)

---

**Signup sheet for OSS project/Writing assignment assignment**

<table>
<thead>
<tr>
<th>Topics</th>
<th>Selections</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic #</td>
<td>Topic name(s)</td>
</tr>
<tr>
<td>E1711</td>
<td>Refactor delayed_mailer.rb and scheduled_task.rb</td>
</tr>
<tr>
<td>E1713</td>
<td>Refactor penalty_helper.rb and late_policies_controller.rb</td>
</tr>
<tr>
<td>E1714</td>
<td>Allow instructor to sign students up for topics and drop them</td>
</tr>
<tr>
<td>E1715</td>
<td>Fix and improve rubric criteria</td>
</tr>
<tr>
<td>E1716</td>
<td>Improve e-mail notifications</td>
</tr>
<tr>
<td>E1718</td>
<td>Review requirements and thresholds</td>
</tr>
<tr>
<td>E1719</td>
<td>Reviewer-round matrix for viewing scores from different rounds</td>
</tr>
<tr>
<td>E1720</td>
<td>Use Ajax for adding participants, TA, Edit Questionnaires screens</td>
</tr>
<tr>
<td>E1721</td>
<td>Test heat map for viewing scores</td>
</tr>
<tr>
<td>E1722</td>
<td>Remove duplicated code in feature tests</td>
</tr>
<tr>
<td>E1723</td>
<td>Fix staggered deadline assignments</td>
</tr>
<tr>
<td>E1724</td>
<td>Remove cache field in roles table</td>
</tr>
<tr>
<td>E1725</td>
<td>Move sign_up_sheet view logic to sign_up_sheet_helper.rb</td>
</tr>
</tbody>
</table>

Figure 19. Students' user interface for bidding feature in Expertiza
In the design of the team formation web service, I provided help and designing the data sharing protocol and the heuristics algorithm (Akbar et al., Submitted.).

Table VII. Usage of PRML Entities in Different Web Services

<table>
<thead>
<tr>
<th></th>
<th>Reputation</th>
<th>Summarization</th>
<th>SimiCheck</th>
<th>Tile Chart</th>
<th>Team Formation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actors</td>
<td>id</td>
<td>id</td>
<td>id</td>
<td>id</td>
<td>id (optional)</td>
</tr>
<tr>
<td>Actor_participants</td>
<td>id (assessors only)</td>
<td>id</td>
<td></td>
<td>id</td>
<td></td>
</tr>
<tr>
<td>Artifacts</td>
<td>id</td>
<td></td>
<td>content</td>
<td></td>
<td>content_case</td>
</tr>
<tr>
<td>Criteria</td>
<td>id</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Critiques</td>
<td>assessor_id</td>
<td>assessor_id</td>
<td>assessor_id</td>
<td>comment</td>
<td>score/rank</td>
</tr>
<tr>
<td></td>
<td>assesseee_id</td>
<td>assesseee_id</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>score (holistic)</td>
<td>comment</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Items</td>
<td></td>
<td></td>
<td></td>
<td>content</td>
<td></td>
</tr>
<tr>
<td>Participants</td>
<td>id</td>
<td>id</td>
<td></td>
<td></td>
<td>id</td>
</tr>
<tr>
<td>Tasks</td>
<td>id</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Additional info. needed</td>
<td>former reputations (optional)</td>
<td>color scheme (optional)</td>
<td>bids by participants and actors</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

From Table VII we find that the PRML can also be used as a data protocol between clients and web service servers in an educational peer-assessment scenario. Among the five web services we have investigated, the majority of data – if not all – are defined in PRML except for the Team Formation web service. The value of PRML in this case, is to share part of data from peer-assessment system to call specific web services. This, again, shows the value of PRML in peer-assessment systems in education.
3.5 Contribution and Conclusion

This section presents our design of data sharing approach based on PRML. To accelerate the queries of shared data and make the data sharing process easier for peer assessment system owners/developers, the star schema of PRML is used. Two examples are given to demonstrate how the ETL project can be undertaken with ETL tools. However, there are still some cases where data sharing (at least partially) requires implementing ETL applications. In this case, the ETL application has to be designed with the collaboration of the PRML designers and peer assessment system owners/developers.

We evaluated the PRML in two different scenarios: using PRML to share full data from peer-assessment systems to the data warehouse, and to share partial data from peer-assessment systems to web services. Our analysis and development based on four peer assessment systems and five web services show that PRML can be used for both purposes. Therefore, we are confident that PRML is a powerful tool which facilitates design and development of further features and data platforms for the entire peer-assessment research community.
CHAPTER 4 Comparison of Reliability of Rating vs. Ranking

4.1 Introduction

The IT revolution has enabled a host of online peer assessment systems, which automate the workflow and remind students to complete their tasks (McLean 2013; De Alfaro and Shavlovsky 2014; Tinapple, Olson, and Sadauskas 2013; E. Gehringer 2009; Babik et al. 2016). The dozens of known educational peer-assessment applications fall into two categories: systems based on rating, and systems based on ranking. Rating-based peer assessment asks assessors to rate each artifact on a Likert scale (or multiple Likert scales for several criteria). Ranking-based peer assessment, on the other hand, asks assessors to rank several artifacts against each other.

In our previous work, we defined a Peer-Review Markup Language (PRML), which models the common entities and relations in the educational peer-assessment process (Song, Pramudianto, and Gehringer 2016). Based on this PRML, we are able to collect data generated by different peer assessment systems and build a data warehouse for the data from multiple systems.

This section describes my efforts to design metrics comparing peer-assessment reliability from data generated by both rating and ranking-based systems.

4.2 Related Work

Advocates of rating and ranking-based systems cite advantages (e.g., the ranking-based assessment shows better robustness across rankers (Tinapple, Olson, and Sadauskas 2013) while rating-based peer assessment better supports detailed rubrics (Song, Hu, and Gehringer 2015a), but no comprehensive quantitative comparison has been done of the reliability of these two approaches. There are multiple reasons for this. First, carrying out this comparison requires data collection and transformation from multiple online systems. Some experiments
have been done on limited data set, e.g., in Love’s experiment, 145 police officers peer-reviewed others in the same squad with both peer-ranking and peer-rating approach (Love 1981); in Shah’s case study, 1879 students’ peer assessment were investigated, but they were all from a single MOOC course (Shah et al. 2013). Though both experiments reported that ranking-based peer assessment is more reliable, the results are still not convincing enough since they only represented one task and one group of peer assessors.

Second, most of the existing measures of reliability (e.g. ICC, correlation coefficient) are suitable only for rating-based systems, and therefore there is no metric of reliability that can be applied to both rating-based and ranking-based peer assessments.

4.2.1 Rating-Based and Ranking-Based Peer Assessment

Peer ranking and peer rating have been long recognized as two main approaches for peer assessment (Kane and Lawler 1978). However, sometimes researchers may confuse peer ranking with peer rating (van Zundert, Sluijsmans, and van Merriënboer 2010). Though arguably they can be done at the same time, they are built into online educational peer assessment systems with clearly different purposes and emphases.

4.2.2 Rating-based Peer Assessment

Rating-based peer assessment systems usually ask assessors (usually student peers, sometimes teaching staff as well (Hamer, Ma, and Kwong 2005; Song, Hu, et al. 2016) to rate artifacts. The assessors usually review one artifact at a time; therefore detailed review rubrics can easily be applied to peer rating because the reviewed artifact holds the assessor's attention well. In practice, teaching staff may design detailed rubric questions to guide assessors in reviewing different aspects of the artifact, and giving either numerical or textual feedback (or both). The aggregated numerical feedback can be considered as the total score that the assessor assigns to the artifact.

Figure 20 is a screenshot from Expertiza (E. Gehringer 2009). Each bullet point is one rubric criterion. Based on each criterion, assessors are asked to give both numerical feedback (on a 5-point Likert scale) and textual feedback. The authors, then, can learn on which aspect
they did well or not so well. If the review rubrics are designed properly, longer and more helpful feedback can also be triggered (Song, Hu, et al. 2016).

4.2.3 Ranking-based Peer Assessment

The designers of ranking-based peer assessment systems usually argue that common understanding of the rating standards is hard to reach (Tinapple, Olson, and Sadauskas 2013), e.g., on a 5-point Likert scale, how does a “3” differ from a “4”? To reach such a common understanding, a calibration or training phase should be applied (Hu, Song, and Gehringer 2016; Song, Gehringer, et al. 2016), which makes the assignment design more complicated.

To avoid the hassle of crafting detailed rubrics and at the same time, improve the reliability of peer assessment, peer ranking is utilized in some systems. The assessors are usually asked to review a fixed number of artifacts. Instead of giving numerical scores to each of the artifacts, assessors need to rank them in order, from strong to weak. Researchers argue that ranking is a more reliable approach to peer assessment because the quantitative feedback (ranking in this case) is given by the comparisons between one artifact and others.
Figure 21 is part of the user interface from Mobius SLIP (McLean 2013) which is a ranking-based peer assessment system. Assessors, in this case, are assigned to review four artifacts from their peers and rank them together with their own artifacts. Each assessor can use the scroll bars to rank those five artifacts. Textual feedback is also supported by Mobius SLIP, but it can only be holistic feedback instead of feedback facilitated by detailed rubrics.

![Figure 21. Screenshot from a ranking-based peer assessment system (Mobius SLIP)](image)

### 4.3 Reliability Algorithm

#### 4.3.1 Global Ranking

After enough peer assessments have been done, it is possible to approximate a global rank for an assignment. This can be done with data generated by either rating-based or ranking-based peer-assessment systems. Some systems may display the global rank on leaderboards (Denny et al. 2008) or simply point students to strong work (Tinapple, Olson, and Sadauskas 2013). The basic assumption of this research is that there is a “ground truth” global ranking of all the artifacts submitted for a specific assignment. This ground truth can be considered as a rank based on the peer assessments received by each artifacts. An artifact will be higher in the global ranking if it is rated/ranked higher than other artifacts assessed by the same reviewers.

To give a precise definition of the algorithm which generates the global rank for an assignment, we adopt the following notations:

- let $a$ be an artifact;
- let $A$ be the set of all the artifacts;
- let $r$ be a reviewer;
- let $R$ be the set of all the reviewers;
- let $r_a$ be the quantitative grade that $r$ assigned to $a$;
- let $R_r$ be the set of artifacts reviewed by $r$;
let \( A \) be the set of reviewers who have reviewed \( a \);
let \( G_s \) be the aggregated grade for artifact \( a \) based on existing peer assessments;
let \( \text{GlobalRank} \) be the global ranking for all the artifacts \( A \) in an assignment;
let \( \text{GlobalRank}_a \) be the global ranking for artifact \( a \).

\( G_s \) can be defined as the average peer-assessment score/rank for artifact \( a \):

\[
G_s = \frac{\sum_{r \in A} g^r_a}{|A|} \quad (4.1)
\]

In the peer-ranking scenario, \( G_s \) is the average rank that artifact \( a \) has received; in the peer-rating scenario, \( G_s \) is the average peer-review score that artifact \( a \) has received.

Based on \( G_s \) for all \( a \in A \), \( \text{GlobalRank}_a \) can be defined as:

\[
\text{GlobalRank}_a = \sum_{a' \in A, a' \neq a} |G_s > G_{s'}| + 1 \quad (4.2)
\]

Adding one is to make the rank start from 1 instead of 0. \( \text{GlobalRank} \) can be considered as a sequence of artifacts in the order of \( \text{GlobalRank}_a \).

Please note that 1) it is possible for multiple artifacts to have the same rank and 2) it does not matter whether the \( \text{GlobalRank} \) is from best to worst or vice versa as long as the \( \text{GlobalRank} \) of the artifacts is maintained.

4.3.2 Tuple-Based Reputation

In peer assessment, the assessments done by a reviewer \( r \) who has reviewed more than one artifact can be considered to define a \( \text{LocalRank}_r \). For example, reviewer \( r \) has reviewed artifact \( a_1, a_2, a_3, \) and \( a_4 \), and \( r \)'s local rank is \( \{a_1, a_2, a_3, a_4\} \) in the same numerical order as \( \text{GlobalRank} \), for example, from high to low. The \( \text{LocalRank}_r \) can be further broken into six (there are four assessed artifacts, so \( C(4, 2) \) possible combinations) 2-tuples. For each 2-tuple, we can define the weight of assessing the two artifacts as \( \text{Weight}(<a, a>) \).

Depending on whether an assessor's assessment for a 2-tuple agrees with the \( \text{GlobalRank} \),
we can define the reliability achieved by this assessing this tuple:

\[
Achieved(<a_i, a_j>) = \begin{cases} 0, & <a_i, a_j> \text{ disagree with GlobalRank} \\ \text{Weight}(<a_i, a_j>), & \text{otherwise} \end{cases}
\] (4.3)

For each assessor \( r \), the reliability can be defined as:

\[
\text{Reliability}_r = \frac{\sum_{<a_i, a_j> \in \text{LocalRank}} \text{Achieved}(<a_i, a_j>)}{\sum_{<a_i, a_j> \in \text{GlobalRank}} \text{Weight}(<a_i, a_j>)}
\] (4.4)

The range of reliability is \([0,1]\).

If we assign all 2-tuples the same weight (to make it simple, we assume a weight of 1), the reliability becomes the percentage of the 2-tuples where the assessor agrees with the \( \text{GlobalRank} \):

\[
\text{Weight}(<a_i, a_j>) = 1
\] (4.5)

We call this algorithm Tuple-Based Reliability-Unified (TBR-U) in the rest of this paper.

If we decide that mistakenly rating/ranking an obviously stronger artifact lower than a much worse one should be punished further, we can modify the (5) as below:

\[
\text{Weight}(<a_i, a_j>) = |\text{GlobalRank}_{a_i} - \text{GlobalRank}_{a_j}|
\] (4.6)

In this case, for a tuple \(<a_i, a_j>\) which does not match the \( \text{GlobalRank} \), the greater difference on \( \text{GlobalRank}_{a_i} \) and \( \text{GlobalRank}_{a_j} \), the lower reliability the assessor will get.

We call this algorithm Tuple-Based Reliability-Differentiated (TBR-D) in the rest of this paper.

The difference between TBR-U and TBR-D algorithm is that TBR-D uses higher weights for tuple \(<a_i, a_j>\) if the difference of \( \text{GlobalRank}_{a_i} \) and \( \text{GlobalRank}_{a_j} \) is larger. For example, consider an assignment with 20 artifacts where \( a_1 \) is the strongest and \( a_{20} \) is the weakest. If an assessor mistakenly assesses those artifacts (considered as tuple \(<a_i, a_{20}>\) in a wrong order,
the decrease in the reliability should be higher than if (s)he mistakenly assesses the tuple $<a_{i0}, a_{i1}>$.

### 4.4 Experimental Results

The experiment was done on data from the PeerLogic data warehouse (F. Pramudianto et al. 2016). The data is taken from multiple educational peer assessment systems and transformed into the same schema. There are 15,498 assessments (assessments give either a rank or comprehensive rating, comprising responses to all criteria in a single rubric) from 466 assignments used in this experiment. Table VIII provides more details of our dataset.

<table>
<thead>
<tr>
<th>Table VIII. Details about the Dataset Used in Experiment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>Num. of assignments</td>
</tr>
<tr>
<td>Num. of participants</td>
</tr>
<tr>
<td>Num. of assessments</td>
</tr>
<tr>
<td>Avg. participants per assignment</td>
</tr>
</tbody>
</table>

Due to different class settings, the courses which used rating-based peer assessment required students to do fewer assessments (2 on average) than the courses which used ranking-based assessment (4 on average).

We calculated the individual reliabilities for all the assessors based on TBR-U and TBR-D algorithms. The distributions of students’ reliabilities are plotted in Figure 22 and Figure 23.

The bars on the right side of Figure 22 and Figure 23 show that more student assessors have reliabilities between 80% to 100% than have lower reliabilities. Comparing the reliability from rating-based assessors, we found that the average reliability from TBR-D (0.690) is almost the same to the average reliability from TBR-U (0.692). For the reliability from ranking-based assessors, we found that the average reliability from TBR-D (0.890) is higher than the average reliability from TBR-U (0.821). Table IX gives more details about the reliability of those two algorithms.
Figure 22. Distribution of assessors’ reliability in rating-based assignment

Figure 23. Distribution of assessors’ reliability in ranking-based assignments

Table IX. Statistics of Assessor Reliabilities

<table>
<thead>
<tr>
<th></th>
<th>Rating-based</th>
<th>Ranking-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. TBR-U</td>
<td>0.692</td>
<td>0.821</td>
</tr>
<tr>
<td>Std. TBR-U</td>
<td>0.314</td>
<td>0.169</td>
</tr>
<tr>
<td>Avg. TBR-D</td>
<td>0.690</td>
<td>0.891</td>
</tr>
<tr>
<td>Std. TBR-D</td>
<td>0.334</td>
<td>0.190</td>
</tr>
</tbody>
</table>

The average reliabilities from rating-based assessors and ranking-based assessors show that the assessors tend to be more reliable in ranking-based assessment no matter which algorithm
is used. On average the ranking-based assessors can rank 12.9% more of the artifacts correctly (or at least agree with the majority of other assessors) compared with the rating-based assessors. We might hypothesize that this is explained by the fact that ranking-based assessors need to compare different artifacts before they give the rank, therefore the ranks are likely to be more reliable. The rating-based assessors, on the other hand, review only one artifact at one time, therefore there is a higher chance that they may rate a weaker artifact higher than a stronger one.

The TBR-D algorithm punishes the assessors more if they mistakenly rate artifacts which are clearly differentiable. In other words, if an assessor rates the best artifact lower than the worst artifact in an assignment (which is more unlikely to happen), the reliability will drop more in TBR-D than in TBR-U. Therefore, we expect the average reliability from TBR-D to be higher than the average reliability from TBR-U. We found this to be true among ranking-based assessors: the average reliability increases almost 7% from TBR-U to TBR-D. This means that though it is harder for assessors to rank two artifacts of roughly the same quality, assessors tend to rank more differentiable artifacts correctly.

However, the rating-based assessors do not show this pattern: the average reliability from rating-based assessors barely changes from TBR-U to TBR-D. This means that the likelihood for rating-based assessors to make mistakes (generating 2-tuples which do not agree with the global rank) is almost the same regardless of the global ranks of two random artifacts. We hypothesize that this is mainly caused by the fact that the assessors do not re-visit the artifacts they have reviewed frequently, especially before they assess new artifacts.

**4.5 Contribution and Conclusion**

We have introduced tuple-based reliability for measuring assessors' reliability in educational peer assessment. This concept can measure reliabilities for both rating-based and ranking-based assessments, and therefore can be used to make a comprehensive comparison on assessors' reliabilities for both peer-assessment settings. Our experiments on the Peerlogic
data warehouse (Song, Pramudianto, and Gehringer 2016) show that the reliability achieved by ranking-based assessors is on average higher than rating-based assessors’. This finding corroborates Shah et al. (Shah et al. 2013) and suggests that if reliability is the higher priority for the teaching staff, ranking-based peer assessment system may be a better choice.

However, the rating also has its advantages over ranking. It is easier to use a detailed review rubric in a rating system because a reviewer can rate an artifact on a set of criteria without flipping back and forth among the artifacts for each criterion on which the artifacts are ranked. In a system with detailed rubrics, authors can receive more formative feedback, which helps them to improve their artifacts (Song, Hu, et al. 2016).

There is also more space for rating-based peer assessment system designers to improve their tools, and facilitating assessors to re-visit their previous assessments should be one of them. The visualization in Figure 21 is from a ranking-based system; however, the designers can also build this kind of visualization into rating-based systems to remind assessors of the artifacts they have already assessed and the scores they assigned to them. We believe features like this can improve the reliability of rating.
CHAPTER 5 Collusion in Educational Peer Assessment

5.1 Introduction

In the peer-assessment process, the key to success is to ensure fair and accurate assessment. Some researchers have attempted to calculate reviewers’ reputations based on their credibility, then use reviewers’ reputations to weight their reviews (Hamer, Ma, and Kwong 2005; Lauw, Lim, and Wang 2007; Song, Hu, and Gehringer 2015b). Another approach which has also proven to be successful is calibration (Russell 2001). The calibration can be done on a few artifacts (either on sample artifacts or real students’ artifacts) with representative mistakes that students may make. Researchers have found that common understanding on rating standards can be better established after calibration (Hamer, Ma, and Kwong 2005).

Those attempts on improving peer-assessment quality work only under the assumption that students are honest and treat the peer assessment seriously. Most researchers simply assume that this is true without paying enough attention the possibility of collusion. This is partially because that there is no formal definition of collusion in peer assessment (Barrett and Cox 2005). Some researchers treat some types of collusion (e.g. pervasive collusion discussed in the next section) as parts of reliability measures, such as “spread” (Cho, Schunn, and Wilson 2006), but we argue that there is a difference between unreliability and collusion: collusion refers to the intended cooperation in peer assessment, either to help some students unfairly gain higher aggregated peer-assessment scores, or to avoid the need to spend serious amounts of time on peer assessments.

Another reason for the dearth of research on students’ collusion in peer assessment is that there is no tool designed to help users of peer-assessment systems, especially teaching staff, to identify potential colluders. As a result, they may have to use “spreadsheet and visual checks” (Fellenz 2006) to detect collusion patterns, which is time-consuming and subjective.

In this chapter, we present our work on defining and investigating students’ collusion in
peer assessment. We found that the collusions inflated the average peer-assessment scores by 1–2%, in addition, instructors should be vigilant against a particular type of collusion. We also developed a tool which can help instructors identify potential colluders in peer-reviewed assignments. This tool has been made available as a web service which can be plugged to different online peer assessment tools.

5.2 Related Work

Collusion among students is a threat to peer assessment validity and fairness that cannot be mitigated by training or calibration. Unfortunately, this issue has not drawn enough attention from researchers in this area. A few researchers have claimed that they have observed this phenomenon, but no formal definition has yet been given on students' collusion in peer assessment. To prevent collusion, some instructors modify the design of their peer assessment, e.g., by preserving anonymity, having reviewers review different artifacts in the second rounds (Ballantyne, Hughes, and Mylonas 2002). In this section, we define and discuss two types of collusion we investigated in this research. Related research and observations are also reviewed and discussed.

5.2.1 Pervasive Collusion (PC)

Topping suggested that some reviewers may "submit average scores, leading to lack of differentiation" (K. J. Topping 2009). However, more often than not, we found students in online peer assessment systems tend to submit high peer-review scores (sometimes even top scores available) (Song, Hu, and Gehringer 2015a). We define this kind of behavior as pervasive collusion. This behavior makes it harder to identify the top artifacts (an artifact of medium quality may still get as good a score as a top artifact because of grade inflation) (Falchikov 2004). From the student's perspective, the main reason for this type of collusion is to reduce the workload and to avoid receiving arguments from the authors.

Even though students may not consciously conspire, teaching staffs have observed that “a nucleus of students starts to assign top grades to all the submissions they review, other
initially honest students see what is happening, and join the colluders” (De Alfaro, Shavlovsky, and Polychronopoulos 2016). Though quantitative analysis was not provided, this observation itself provides an important reason to investigate this type of collusion. In the worst case, if a higher percentage of students join this activity and give/receive (free) high peer-review scores, the honest reviewers could be penalized as outliers, since they disagree with their “colluding peers.”

5.2.2 Small-circle Collusion (SCC)

Sometimes as we monitored students’ behavior in online peer assessment systems, we found that some students gained an unfair advantage by giving higher peer-review scores to students they know. The reason is usually that students value personal qualities such as friendliness and trust more than mandates on academic conduct (Sutherland-Smith 2013). We define this kind of behavior as small-circle collusion. This is, to some extent, deliberate cheating. Another alternative is to consider small-circle collusion as "friendship bias" (Love 1981)—friends, if somehow able to figure out which is their friends' work, can give their friends higher scores even in blind review.

Though the average peer-review scores may not be used as part of students’ final scores for their artifacts, they may still influence the teaching staff when deciding the final scores. From students’ perspective, the main reason for small-circle collusion is to help friends. Similarly, "to help friends" is also known as one of the principal reason for students dishonest behaviors including plagiarism, cheating, and falsification (Franklyn-Stokes and Newstead 1995).

5.3 Detection of Potential Collusions

In this research, we mainly focused on two research questions related to collusion in peer assessment in education.

RQ 1: How much inflation can be created by collusion?

Since some educators may aggregate peer-review scores into final scores for each artifact,
the influence of peer assessments by colluders should be removed or at least mitigated. Unfortunately, the functionality of identifying and removing colluders’ peer assessments is usually unavailable in online peer-assessment systems. In this case, if the teaching staff simply use the average or median of the peer assessment scores as the final scores, there could be a potential inflation of students’ scores.

Even when the teaching staff does grading completely independently of the peer-assessment process (teaching staff grade all the artifact by themselves without even referring to the peer assessments), there is still another inflationary effect from the authors’ perceptions — high peer-assessment scores may give them the illusion that their artifacts are already of good quality, and therefore they could be less willing to make modifications. They might even ignore the honest (yet helpful) advice.

RQ 2: If, to some extent, there is collusion and inflation in peer assessment, which approach, ranking or rating, can help lower the effect of collusion?

Peer ranking and peer rating have been long recognized as two main approaches for peer assessment (discussed in Section 4.2). Ranking-based peer assessment systems usually ask the reviewers to review a fixed number of artifacts. Instead of giving numerical scores to each of the artifacts, assessors need to rank them in order, from strong to weak. Researchers argue that ranking is a more reliable approach to peer assessment because the quantitative feedback (ranks in this case) is given by the comparisons between one artifact and others (Tinapple, Olson, and Sadauskas 2013).

Some comparisons between ranking-based and rating-based systems have been done, e.g., on review reliability (Song, Yifan, and Edward 2017). However, there has been no comparison of collusion in peer-rating and peer-ranking settings. Since collusion among students is a major concern of some educators (Falchikov 2004), it is important to know how to design the peer assessment to minimize the potential collusion.
5.3.1 Dataset

This study was conducted with the PeerLogic\(^2\) dataset. The data is derived from multiple educational peer-assessment systems used by college students in a variety of disciplines including computer science, mechanical engineering, electrical engineering, statistics, art and English (Ferry Pramudianto, Aljeshi, et al. 2016). The shared data have been transformed and combined to facilitate further research (Song, Pramudianto, and Gehringer 2016). There are 45,991 assessments (assessments give either a rank or comprehensive rating, comprising responses to all criteria in a single rubric) from 309 assignments used in this experiment. Table X provides more details about our dataset. Due to different class settings, the courses which used rating-based peer assessment required students to do fewer assessments (2 on average) than the courses which used ranking-based assessment (4 on average).

<table>
<thead>
<tr>
<th></th>
<th>Rating-based</th>
<th>Ranking-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Num. of tasks</td>
<td>168</td>
<td>141</td>
</tr>
<tr>
<td>Num. of participants</td>
<td>1519</td>
<td>1366</td>
</tr>
<tr>
<td>Num. of assessments</td>
<td>21343</td>
<td>24648</td>
</tr>
<tr>
<td>Avg. participants per assignment</td>
<td>50.6</td>
<td>46.9</td>
</tr>
</tbody>
</table>

All the peer assessments in our dataset were done in double-blind fashion.

5.3.2 Collusion Detection Algorithms

The peer assessment in each task can be considered as a directed graph, in which each student (assessor or assessee) can be considered as a vertex and each review can be modeled as an edge from the assessor to the assessee. The weight of each edge can be considered as the numerical peer-review score (either a ranking or a rating score). To give a precise definition of the two types of collusion discussed in section 5.2 we adopt the following notations:

- Let \( G = (V, E) \) be a directed graph for a peer-assessment task,
\( V \) be all the vertices in this graph (namely, all the students who have participated in the peer assessment); and \( E \) be all the edges (namely, all the peer assessments done by students).

A path in \( G \) is a sequence of vertices \( p_{uv} = (v = v_1, v_2, \ldots, v_k = u) \) such that \((v_i, v_{i+1}) \in E\) for \( 1 \leq i < k \). A circuit is a path where \( v \) and \( u \) are identical, \( C_{uv} = (v = v_1, v_2, \ldots, v_k, v_1 = v) \).

We say that an edge is in a circuit, \( e \in C_{uv} \), if \( e \in \{(v_k, v_1) \cup (v_i, v_{i+1}) \} \) for \( 1 \leq i < k \); a vertex is in a circuit if \( u \in \{v_1, v_2, \ldots, v_k\} \).

We assume that there is no single-vertex circuit (one reviews him/herself, a.k.a self-review) in this directed graph. We also assume that there can be only 0 or 1 edge between any pair of vertices (one assessor can only give one aggregated numerical score to a particular assesseee).

Let \( E_{\text{in}}^v \) be all the edges directed to vertex \( v \)

Let \( E_{\text{out}}^v \) be all the edges directed from vertex \( v \)

Let \( W_e \) be the weight of an edge \( e \) (in this case a weight is a peer-assessment score)

Let \( \alpha \) be the threshold of the weight to be considered as high enough to indicate potential collusion

Let \( \beta \) be the threshold size of small-circle collusion circuits.

A potential small-circle collusion can be formally defined as:

\( C_{uv} \) is a circuit in \( G \),

For each \( e \in C_{uv}, W_e > \alpha \),

\( k \leq \beta \),

\( \exists u \in C_{uv} \) that makes \( \overline{W}^{in}_{\text{within } C} / \overline{W}^{in}_{\text{outside } C} \geq 1 + \varepsilon \), in which

\[
\overline{W}^{in}_{\text{within } C} = \frac{\sum_{e \in E_{\text{in}}^u, e \in C} W_e}{|e|_{E_{\text{in}}^u, e \in C}} \tag{5.1}
\]
The first part of the formal definition of potential small-circle collusion is simply that the suspects need to be a circuit in the directed graph of the peer-assessment task. The second part indicates that the peer assessments in the potential small-circle collusions should have higher numerical scores than the threshold $\alpha$. This will filter out the circuits of students who give each other medium or low scores. The third part of this definition adds a constraint on the number of vertices of the circuits − the circuits with more than $\beta$ vertices are less likely to become a “small”-circle collusion. The last part of this definition emphasizes that there should be peer-assessment disagreement between the peer-assessment scores within and outside the circuits.

Figure 24 shows a visualization of a potential small-circle collusion in a peer-assessment task. In this visualization, each vertex is a student in the task. Each directed edge represents a peer assessment from the assessor to the assessee. In this case, all the peer-assessment scores are higher than the threshold $\alpha$ (edges are in green) and the number of vertices (3) is smaller than $\beta$. However, we are still not confident that those three students are possibly colluders because they could be three of the strongest students in the class who happened to review each other.

Figure 25 shows the peer assessment received by one of the three students. Each red edge represents a peer assessment with a score lower than $\alpha$. From the visualization, we found that, though the student “a*****1” received two favorable peer assessments from the other two students who are in the potential small collusion circuit, (s)he overall received fewer favorable peer assessments than unfavorable ones. This indicates there is a disagreement between the reviews inside and outside the circuit. If the difference is greater than the threshold $\epsilon_1$, our algorithm will consider this circuit as a case of potential small-circle collusion.
A potential pervasive-colluder vertex $u \in V$ can be formally defined as:

- $E_{\text{out}}^u \geq \text{Avg}(E_{\text{out}}^v)_{v \in V}$,
- $|e|_{e \in E_{\text{out}}^v, W_r > \alpha} / |E_{\text{out}}^u| \geq 1 - \epsilon_2$. 

Figure 24. An example of a small-circle collusion

Figure 25. Disagreement between peer-assessment scores within and outside the small-circle detected
The first part of the definition makes sure that only the vertices with $E_{out}^v$ greater than the average will be considered. This is to exclude reviewers who reviewed only a small number of artifacts that happened to be the stronger ones. The second part of the definition examines the percentage of peer-review scores lower than $\alpha$ given by this reviewer: if it is lower than $E_2$ (in other words, more than $100\%-E_2$ of the reviews given have peer-review scores higher than $\alpha$), this reviewer will be considered to be a potential pervasive colluder.

Figure 26 is another visualization of the same peer-assessment task. In this case, reviewer “f******i” assessed 11 artifacts, which is higher than average. In addition, all the peer-assessment scores assigned were higher than $\alpha$ (green edges). The combination of the high number of reviews done and high percentage of scores higher than $\alpha$ made him/her a suspect for being a pervasive colluder.

![Figure 26. An example of a pervasive colluder](image)

Please note that our algorithm only detects the “suspect” colluders in our dataset. We do not have any further evidence since the dataset used in this research is anonymized and from different peer-assessment systems.

Our collusion detection for both types, and the visualization will be made available as a
web service. This web service accepts data in PRML (Peer-Review Markup Language) format (Song, Pramudianto, and Gehringer 2016) and can detect and visualize potential collusions in any peer assessment task.

5.4 Experiment Results

In our experiment, we set $\alpha$ to be the top 80% quantile of the peer-assessment scores for each task, namely, a peer-assessment is considered to be favorable if it is higher than 80% of the peer-assessments in a task. $\beta$ was set to 4 and both $\epsilon_1$ and $\epsilon_2$ were set to be 5%. Those parameters are predefined by Expertiza research group and the instructors who have used Expertiza system.

5.4.1 Pervasive Collusion

We created an algorithm to detect pervasive colluders and calculate the percentage of reviews by pervasive colluders in each task. Figure 27 shows the distribution of those percentages (x-axis shows the percentages of reviews done by pervasive colluders, the interval between each bin boundary is 5%; y-axis shows the number of tasks). We found that in more than 80% of tasks, less than 10% of reviews were submitted by pervasive colluders. Only very few tasks had 50% or more reviews done by pervasive colluders (will be discussed in the next section).

Figure 27. Distribution of pervasive colluders’ reviews percentages
5.4.2 Small-circle Collusion

We also computed the percentage of reviews done within small collusion circles for each task. Those small-circle collusions were detected with Hawick and James’ algorithm (Hawick and James 2008) with necessary modifications, e.g. parameters such as $\alpha$ and $\beta$. Figure 28 illustrates the distribution of those percentages (the $x$-axis shows the percentages of reviews done by small-circle colluders, the interval between each bin boundary is 5%; and the $y$-axis shows the number of tasks). We found that more than 70% of tasks contain 5% favorable reviews or less from small-circle collusions. Very few tasks had more than 15% of reviews from small collusion circles.

![Figure 28. Distribution of small-circle colluders' reviews percentages](image)

5.4.3 Score Inflation Brought by Collusion

We removed all the reviews done by pervasive colluders and small-circle colluders and calculated the average peer-assessment scores for each task. We further compared those average scores with the original average scores (without removing colluders’ reviews). The scores from ranking based system are scales from $100/n$ to 100, where $n$ is the number of artifacts ranked by each assessor (Tinapple, Olson, and Sadauskas 2013).

By comparing two average review scores for each task, we can estimate the score inflation brought about by collusion. Figure 29 is the distribution of score inflation for all the tasks in our dataset (the $x$-axis shows the score inflation based on 100 points, the interval between each bin boundary is one point, and the $y$-axis shows the number of tasks). This distribution indicated that for more than half of the review tasks, the inflation caused by
colluders was less than 1 point. However, in the worst case, both type of collusion, together could bring almost 5 to 10 points inflation.

Figure 29. Distribution of score inflations brought by collusions

5.4.4 Rating vs. Ranking

We also computed the collusion results and score inflation of data generated by ranking-based reviews and rating-based reviews separately (see Table XI). We found that the average inflation in rating-based peer-assessment scores is slightly higher than the average inflation of ranking-based peer-assessment scores. There is not a big difference between the percentages of small-circle colluders’ review in ranking (3.4%) and rating (3.7%). However, we did not find any pervasive colluders in ranking-based assessment, but we found 9.98% of the assessments done in the rating-based scenario were done by pervasive colluders. This indicates that in both ranking-based and rating-based assessments, students may return favors to each other. Small-circle collusion is the major collusion type for ranking-based peer assessment. However, in rating-based peer assessment, pervasive collusion is an even bigger challenge for educators: we found that on average almost 10% of the reviews in the rating-based system were done by pervasive colluders.

We compared the score inflations introduced by colluders for each assignment between ranking-based and rating-based settings. The \( t \)-ratio of score inflation on rating-based to ranking-based peer assessments is 2.39 with \( p \)-value of 0.0175 (< 0.05). This means that collusion is likely to cause more inflation in rating-based peer assessment. We hypothesize that the main reason is that the ranking-based peer assessment better facilitates comparisons.
It is harder for a ranking-based reviewer to collude since all the artifacts are ranked against each other. Rating-based peer assessment relies less on comparison. As a result, it is easier for colluders to give their friends high/top scores (small-circle collusion), or give each assigned artifact the same score (pervasive collusion).

<table>
<thead>
<tr>
<th></th>
<th>Rating-based</th>
<th>Ranking-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg. inflation</td>
<td>1.42</td>
<td>1.03</td>
</tr>
<tr>
<td>Std. dev. inflation</td>
<td>1.51</td>
<td>1.26</td>
</tr>
<tr>
<td>Avg. SCC %</td>
<td>3.7%</td>
<td>3.4%</td>
</tr>
<tr>
<td>Avg. PC %</td>
<td>10.0%</td>
<td>0.0%</td>
</tr>
</tbody>
</table>

Our experimental results indicate that collusion happens in peer-assessment tasks, and in some cases, the inflation cannot be ignored. In this section, we discuss additional questions related to our results.

5.4.5 Relationship between Pervasive Collusion and Small-Circle Collusion

We argue that pervasive collusion and small-circle collusion are not independent, but rather, if there is a large number of pervasive colluders in a peer-assessment task, it is also likely that our algorithm will find more cases of small-circle collusion. In this scenario, more pervasive colluders increased the numbers of blindly-assigned high scores (or even top scores). If two pervasive colluders happened to review each other, a small collusion circle is likely to be detected. In other words, a higher percentage of pervasive colluders is likely to increase the number of small collusion circles, though those students may instead be pervasive colluders.

An even worse result from a high pervasive collusion rate is that it makes the peer assessment more lenient, even for reviews from the students who are not detected as colluders. In our data set, the average peer-assessment score for non-colluding students on the 10 tasks with highest pervasive collusion rate is 95, yet the average peer-assessment score for non-colluding students is 85 for all tasks. This indicates that students, when receiving more favorable peer assessments, can also be more lenient when reviewing others. We
hypothesize that this is caused by the “peer pressure” from the colluders in the peer-assessment task.

On the other hand, small-circle collusion does not induce “peer pressure” to encourage students to be lenient. We calculated the average peer-assessment scores from non-colluding students on 10 tasks with the highest percentage of small-circle collusions. Though there are many small collusion circles, the average peer assessment scores from honest students in those tasks was 86.7, which is almost the same as the average peer-assessment scores from all the honest students in the tasks (86.4). Hence, we suggest that teaching staff pay more attention to pervasive collusion in peer assessment because they brought more “hidden” inflation to peer-assessment scores (an increase in scores by honest peer reviewers rather than colluders), which is harder to estimate.

5.4.6 How does collusion influence reputation systems?

Reputation systems estimate the credibility of each reviewer in peer-assessment activities. Reputation systems may take various factors into account, but in this section, we only discuss reputation systems based on levels of agreement in peer-assessment scores (Song, Hu, and Gehringer 2015b) in this section. This type of reputation algorithm calculates reviewer reputations and aggregate scores for artifacts recursively until they reach convergence (Hamer, Ma, and Kwong 2005; Lauw, Lim, and Wang 2007).

For small-circle collusion in which colluders give each other higher scores to each other within the circles, if the reviewers outside the circles rate/rank an artifact much lower, the reviewers in the circle are likely to get lower reputation scores. This means that the small-circle colluders tend to be punished by reputation system since their reviews do not agree with the majority of other reviewers on the same artifacts (Song, Hu, and Gehringer 2015b). Therefore, teaching staff does not need to worry much about small-circle collusion if there is a reputation system in the peer-assessment system.

The pervasive colluders, if they are not the majority, may also be punished by reputation systems for the same reason. However, in the worst case (when there are more than 50% of
the reviews are from pervasive colluders, as in 4 tasks from our dataset), the reputation system will fail to detect pervasive colluders since they are not the minority anymore. Instead, all the pervasive colluders will receive higher reputations than the honest reviewers, which could potentially hurt the honest reviewers and lead them to become pervasive colluders as well. This suggests that reputation systems can potentially punish the pervasive colluders by giving them low reputation scores only when they are not the majority.

5.4.7 Will collusions increase during the semester?

Although on average, collusion does not cause major inflation in peer-assessment scores, teaching staff may still worry that more students will learn to collude as the semester progresses (De Alfaro, Shavlovsky, and Polychronopoulos 2016). From our dataset, we pick the courses which have at least two peer-assessment tasks and visualized their inflations against the day of the semester that the peer assessment is due (Figure 30).

![Figure 30. Peer-assessment inflations based on their deadline of day of semesters](image)

In Figure 30, the x-axis shows on which day of the semester that the peer-assessment tasks were due, and the y-axis shows the inflation in the tasks. The slope of the fit line is 0.005. Though it is positive, the slope is relatively small considering the total score in our dataset is 100. In addition, the $R$-square value is 0.01 (the change in the $x$ value only explains 1% of the change in the $y$ value), which is again, very small.

This finding, in general, disagrees with de Alfaro’s observation (De Alfaro, Shavlovsky,
and Polychronopoulos 2016). Instead, we found that students’ collusion changes little within a semester. However, we did find a few cases where the inflation kept increasing in a course through a semester, e.g. in the first peer-assessment task in a course (due on 28th day of the semester), there was 0 inflation but on the third task (due on 99th day of the semester) the inflation was 5.7 points.

5.5 Contribution and Collusion

Educators apply peer assessment to provide authors timely and helpful reviews. Collusion, however, can undermine the value of peer assessment. In this chapter, we identified two types of collusion, pervasive collusion and small-circle collusion, and gave formal definitions for both. We designed algorithms to detect potential colluders of both types with our PeerLogic dataset, and investigated the impact of collusion on peer-assessment scores. We found that on average the collusions inflated the average peer-assessment scores by 1–2%.

We also compared the collusion in rating-based and ranking-based systems. Generally speaking, rating-based peer assessment tends to have more collusion than ranking-based peer assessment. We hypothesize that the main reason is that the ranking-based peer assessment better facilitates comparison and thereby makes students less likely to collude, namely, rank a weak artifact higher than a stronger one. In rating-based peer assessment, since students have fewer chances to do comparisons, they are slightly more likely to collude. Our experimental results on the dataset from different peer-assessment systems show that in most cases, only a small portion of reviews are done by colluders. However, in a few extreme cases, more than 50% of reviews are done by colluders.

We also suggest that teaching staff pay more attention to pervasive collusion (to help instructors, we will make the visualization and collusion detection tool a web service). First, pervasive colluders tend to influence other honest reviewers and make them more lenient than average. This makes it harder to differentiate stronger artifacts from weaker ones, and tends make authors feel overconfident of their work. This harms the peer-assessment process.
since authors do not get many honest/helpful reviews. Second, the pervasive colluders, if they become the majority, will deceive a reputation system and make the honest reviewers outliers. In this case, the opinions of the honest reviewers may be overwhelmed by the high scores given by colluders.
CHAPTER 6 Summary and Future Work

6.1 Summary

This dissertation summarizes our work on collecting and analyzing data from different educational peer-assessment systems. The objective is to provide a data-sharing protocol for the peer-assessment research community and enhance our understanding of how to design peer assessment task in educational settings.

In Chapter 2, we presented Peer-Review Markup Language (PRML) which defines common concepts and relationships between different concepts in educational peer assessment settings. This PRML was designed jointly by a group of peer-assessment system designers/owners. The first PRML was a relational version. Based on this relational PRML, we further designed PRML with a star schema, which better facilitates data sharing between different peer-assessment systems.

Chapter 3 discusses our approach to collecting data from different peer-assessment systems. This project was facilitated by the start schema of PRML. With ETL (Extract, Transform, Load) tools and/or applications developed, we managed to have two systems sharing their data with us and another two have promised to share with us later. The shared data is stored in a data warehouse with read-only access open to the public. This data warehouse can serve as a common data repository for researchers to test their hypotheses and apply data mining to determine better designs for educational peer assessment.

Chapter 4 presents our investigation on comparing peer assessment reliability between ranking-based and rating-based peer assessment. This is done based on the data we have collected. We designed algorithms to calculate reviewers’ reliability based on the global ranking of all the artifacts reviewed in the same task. These algorithms can calculate the reliabilities of reviewers in both ranking-based and rating-based systems.

Experimental results show that the reliabilities of reviewers in more than 200 ranking-
based assignments are on average more than 10% lower than the reliabilities achieved by reviewers in more than 200 rating-based assignments. What is more surprising, the reviewers in ranking-based systems did better on differentiating artifacts of significantly different quality than reviewers in rating-based systems. This suggests that when reviewers are asked to rate artifacts one by one, facilitating revisiting previously rated artifacts should be an important system design to improve the review reliability.

Lastly, Chapter 5 presents our research on students' collusion in educational peer assessment. Collusion refers to students giving each other higher (or even perfect) scores to avoid the need to spend much time in peer assessment. Previous researchers did not pay serious attention to collusion among students, partially because there was no tool available for instructors to do so. For example, they may have had to use "spreadsheet and visually checks" (Fellenz 2006) to detect the collusion patterns, which is time-consuming and subjective.

In this research, we have defined different types of collusions: pervasive collusion and small-circle collusion. Small-circle collusion is the type of collusion where student reviewers gave each other higher assessments; pervasive collusion, on the other hand, refers to the phenomenon that some reviewers give high assessments to every artifact they are assigned to review, which is not necessarily based on conspiracies between a few students.

We can consider the peer-assessment records in one review task as a directed graph in which students are nodes and each assessment is an edge (with scores or ranks as the values of the edges). Therefore, collusion detection can be modeled as a graph-mining problem: detection of small collusion circles can be considered as detection of strongly connected components (with high values on each edge) and detection of pervasive colluders can be considered as detection of nodes from which there are high percentages of edges with high values.

By detecting those possible collusions and remove them from the data set, we found that the colluders bring 1–2% of score inflation to peer-assessment tasks on average. In addition, our experimental results suggest instructors be aware of pervasive collusion since this type of
collusion is more likely to inflate the peer-assessment scores. Compared to ranking-based peer-assessment, rating-based peer assessment is more likely to suffer from collusion.

6.2 Further Work

- **Peer assessment settings other than rating-based and ranking-based.** Ranking-based and rating-based have long been recognized as two main categories of peer-assessment setup. This dissertation has done comparisons on two dimension: reliability and vulnerability to potential collusions. However, there are also other approaches to facilitate assessments, such as peer-nomination (Love 1981).

One of the reasons that peer-nomination is not used in any known peer-assessment system is that peer-nomination does not provide scalable assessments to all the artifacts. For example, an artifact with top quality could be nominated as the best artifact reviewed by many reviewers; an artifact with medium quality may also have chances to be nominated, depending on the artifact it is compared with; but the weak artifacts are not likely to be nominated by any reviewer. The score aggregation, in this case, can help teaching staff to identify the strongest artifact(s), but it is possible that a large portion of the artifacts, if not nominated by any reviewer, will receive the same score. This situation can be potentially solved by allowing reviewers to nominate both the strongest and the weakest artifacts.

Since peer-nomination can be considered as a special case of peer-ranking, peer-nomination can still be a promising peer-assessment approach because it is likely to inherit the known advantages from peer-ranking, e.g., high reliability and less collusion. We could hypothesize that the top artifacts based on the peer-nomination approach should likely be the best (high inter-rater reliability); in addition, students should have less likely to nominate their friends’ artifacts to be the bests if they are clearly not.

- **Extend PRML to include more data with finer granularity.** In PRML, we have
modeled the most common data from several peer-assessment systems. One of the extensions we may consider is to include finer-grained data, such as system logs and the modification history of both artifacts and reviews.

The system logs can give researchers more information on when students submit, review or perform other activities in peer-assessment systems. With this information, researchers can study students’ motivations when giving/receiving peer-assessments. This will give answers to the research questions such as “in peer assessment, how motivated are the reviewers to give feedback?” and “after receiving peer assessment, how motivated are student authors to improve their artifacts?”

Another type of data with finer granularity is the change history of the students’ artifacts (if authors are allowed to make modifications) and reviews (if reviews are not final). With natural language processing (NLP) algorithms, we should be able to find what percentages of the changes made by the authors are suggested by the reviewers.

In this report, we have explained our approach to building data sharing protocol for the educational peer-assessment research community. We used our protocol to facilitate data sharing for four systems and five web services. Based on the data collected from different systems, we have provided a comprehensive comparison of the reliabilities of review between ranking-based assessment and rating-based assessment. We have also defined two types of collusion in peer assessment and studied their impact on educational peer assessment.
REFERENCES


Australia, Australia: Australian Computer Society, Inc.
http://dl.acm.org/citation.cfm?id=1082424.1082433.


https://repository.lib.ncsu.edu/bitstream/handle/1840.16/10975/etd.pdf?sequence=2.


http://www.imsglobal.org/specs/ltiv2p0/implementation-guide.


http://liris.cnrs.fr/alain.mille/enseignements/Ecole_Centrale/What%20is%20an%20ontology%20and%20why%20we%20need%20it.htm.


http://www.biochemsoctrans.org/content/36/1/33.abstract.


