Leveraging Peer Feedback to Improve Visualization Education[1][2]

Zachariah Beasley
zjb@mail.usf.edu
University of South Florida

Alon Friedman
alonfriedman@usf.edu
University of South Florida

Les Piegl
lespiegl@mail.usf.edu
University of South Florida

Paul Rosen
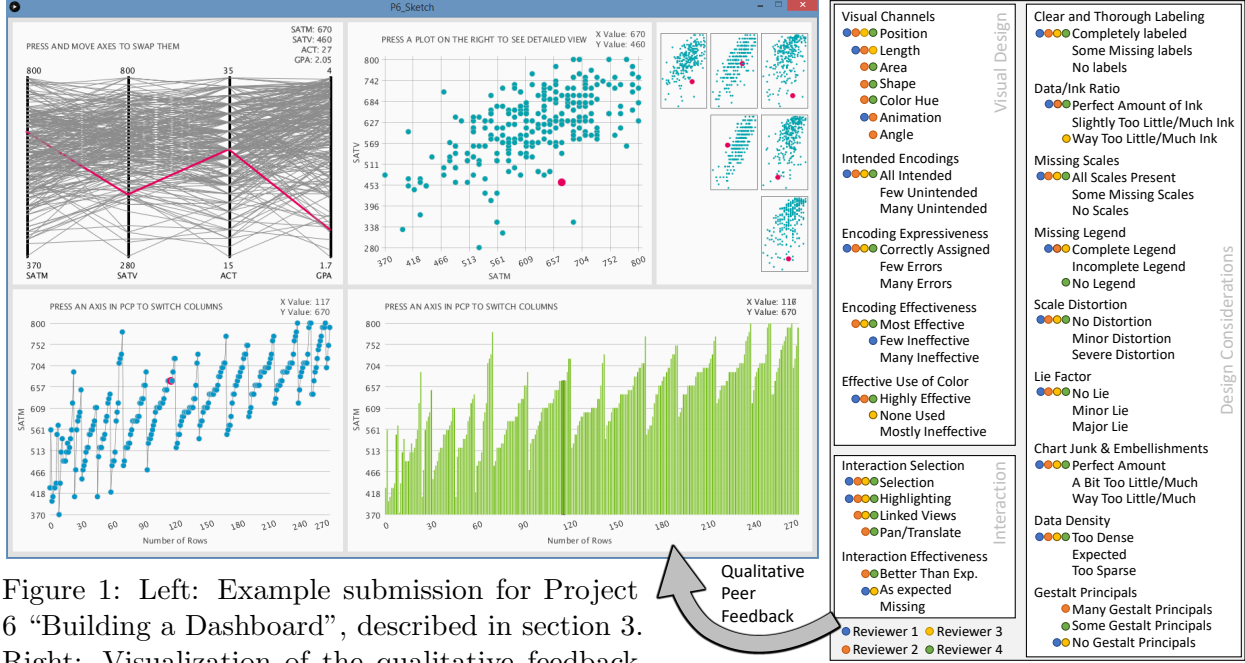prosen@usf.edu
University of South Florida

Figure 1: Left: Example submission for Project 6 "Building a Dashboard", described in section 3. Right: Visualization of the qualitative feedback received from 4 peers to the project submission based upon the rubric described in section 5.

Abstract. Peer review is a widely utilized pedagogical feedback mechanism for engaging students, which has been shown to improve educational outcomes. However, we find limited discussion and empirical measurement of peer review in visualization coursework. In addition to engagement, peer review provides direct and diverse feedback and reinforces recently-learned course concepts through critical evaluation of others' work. In this paper, we discuss the construction and application of peer review in a computer science visualization course, including: projects that reuse code and visualizations in a feedback-guided, continual improvement process and a peer review rubric to reinforce key course concepts. To measure the effectiveness of the approach, we evaluate student projects, peer review text, and a post-course questionnaire from 3 semesters of mixed undergraduate and graduate courses. The results indicate that course concepts are reinforced with peer review—82% reported learning more because of peer review, and 75% of students recommended continuing it. Finally, we provide a road-map for adapting peer review to other visualization courses to produce more highly engaged students.

---

1 Introduction

Rushmeier et al. [42] defined visualization education as a work in progress. Nevertheless, the subject can be broadly split into 2 categories. The first is the proper construction of visualizations—using the right algorithms and visualization principles in creating visualizations, which tends to be the primary focus of visualization courses, and student comprehension of concepts, techniques, and algorithms can be objectively measured. The second category is focused on the subjective evaluation of the quality and accuracy of visualizations. Subjective evaluation is not only important for the instructor's assessment of students but for students to *develop the ability to evaluate others' visualizations critically.* These skills are commonly taught through informal methods, such as group or whole-class discussions that can leave students' skills underdeveloped [43].

Furthermore, with Gen Z learners (individuals born between the mid-1990s and mid-2000s), educational preferences have shifted significantly. They prefer instant feedback, are increasingly collaborative, and are active-learners who prefer project-based coursework [40]. Meanwhile, with a large number of students enrolled in visualization courses, it is difficult to provide students the timely, subjective feedback they need to improve the quality of their work [47].

Peer review is a highly-engaging feedback mechanism [22, 57], often used in liberal arts courses [4, 36, 55, 30], human-computer interaction (HCI) courses [46, 12, 21], code review [1], and scholarly publication [56], and it is ideal for addressing these challenges. Instead of relying solely on instructors for feedback, peers collaborate to provide diverse multi-sourced feedback to one another with a relatively quick turnaround. In addition, the evaluation process itself gives students an opportunity to reinforce recently-learned course concepts by critically evaluating others' work. Finally, for instructors, it is a scalable approach. Since students provide feedback to one another, adding students to a course only adds to nominal administrative efforts.

Despite these well-known advantages of peer review, we found little evidence of peer review in visualization courses. In a survey of 100 information visualization faculty, we found 18 publicly available course syllabi, only 1 of which mentioned peer review. *In order to address this gap, we discuss the construction and evaluation of our own peer review-oriented computer science visualization course, in order to encourage the visualization community to initiate discussions around and to eventually adopt this pedagogical methodology into their classrooms.*

The contributions of this paper are:

1. We describe a peer review-oriented visualization course with projects designed to reuse code and visualizations in a feedback-guided continual improvement process;

2. We discuss and evaluate a peer review rubric to support the continual improvement process while reinforcing key concepts of the course;

3. We evaluate whether peer review reinforces course material, whether students engage in and enjoy peer review, and what aspect of peer review is most beneficial to students; and

4. We discuss various ways to integrate peer review into existing visualization courses.

2 Background

Peer review, the evaluation of work by individuals of similar competence to the producer(s) of the work [48], has been successfully utilized in many professional practices, e.g., code review [1], evaluation of scholarly and scientific works [56], etc. In code review, it serves as a software quality assessment activity. In science, peer review is the method by which papers are published, academic promotions secured, and Nobel prizes won. Notably, in science, the practice has been criticized as

repetition based on faith rather than fact [6].

2.1 Educational Values of Peer Review

Seeing Others' Work In engineering education, the use of peer review in the classroom remains an open area of research [2], despite the fact that it is so essential to the design and implementation of engineering systems. For example, Garousi [15] constructed a framework using a "Goal-Question-Metric" (GQM) approach and reported that students gained knowledge in preparing software development documentation and writing code and that *students find peer review useful primarily since they can learn from their peers' work*. He also suggested that students might benefit from reviewing work from prior semesters before embarking on their projects.

Formative vs. Summative Value Many educators have employed both *formative* reviews to stimulate learning and *summative* reviews for assessment purposes (i.e., providing a grade) in their courses. In liberal arts, multiple researchers have explored whether the ability to write well is a skill that can be mastered in one context and simply transferred to another [4, 36, 55]. Moxley [30] opposed this, stating that grand claims about student ability based on a handful of rubric scores are not necessarily predictive of their classroom performance. This concern applies when considering peer review for *assessment* in visualization courses, though it does not question the *formative* benefits of peer review.

Active-Learning Naps et al. [34] suggested that no matter how well a visualization is designed, there is little educational value unless it engages students in an active-learning classroom activity. Perhaps most notably, peer review is a highly-engaging, active-learning educational mechanism [22, 57], and active-learning has been shown to significantly improve retention, comprehension, and overall educational outcomes [29].

2.2 Visualization Education

There is a wide breadth for the "style" of visualization courses that usually match the focus of the visualization sub-community (i.e., VAST, InfoVis, SciVis), which the instructor primarily subscribes to. For a broad survey of visualization course styles, see [23, 37]. Nevertheless, much of the recent innovation in visualization education has focused on developing problem- or design-oriented courses that apply a senior project style course design to address a visualization problem over an entire semester [19, 58, 45, 41]. In some cases, these include the additional challenge of being interdisciplinary [9, 11, 10].

Course Topics Proposals for formal visualization education have focused mainly on the *proper construction of visualizations and other technical aspects*, not the critical evaluation of visualizations. Hanrahan and Ma [18] proposed the implementation of core topics to visualization education that included: data and image models; perception and cognition; interaction; space; color; shape and lines; texture; interior structure with volumetric techniques; and narrative with animation. Gilbert [17] further described 5 levels of competency—representation as depiction; converting early symbolic skills; the syntactic use of formal representation; semantic use of formal representations; and reflective, rhetorical use of presentation—that give insight to the depth of topical understanding in visualization. More recently, Munzner's *Visualization Analysis & Design* [33] takes visualization concepts and splits them into what-why-how questions that she maps into data abstraction, task abstraction, and visual encoding and interaction technique taxonomies. Our course follows this text.

Visualization Design   A design methodology is critical in activities from software development to visualization design. On the software design side, Agile-style iterative design [5] and the Model-View-Controller design pattern help to develop software engineering skills. One of the more popular methodologies to bridge the gap between software and visualization design, Munzner's Nested Model [32] adapts Agile principals to the specific needs of visualization. The Design Activity Framework further connects the Nested Model to visualization design [28] with worksheets that guide the would-be designer [27]. These approaches are invaluable for teaching composition and design in a visualization course.

A different approach to design is heuristics, "an approach, a strategy, or a trick that experience has shown may help in constructing a useful model" [39]. Santos et al. discussed heuristic evaluation as a usability inspection method that has been adapted to evaluate visualizations [43]. Various researchers have examined student performances in their visualization classes based on heuristic evaluation [20, 54], and the methods provide a potential alternative design evaluation methodology that fits within the context of peer review.

Peer Review in Visualization   Critiquing has long been acknowledged as a critical part of the visualization design process [24], which has, by-and-large, been taught through informal methods. *We have found little evidence that the community has broadly considered peer review as an effective pedagogical method.* We surveyed visualization course syllabi from the websites of 100 professors in the IEEE InfoVis 2019 program committee, IEEE VIS 2019 conference committees, and 2016 and 2017 Pedagogy of Data Visualization IEEE VIS Workshop contributors. Of the 18 syllabi openly available, only 1 mentioned peer review of another student/group's work[3]. Four syllabi suggested at least 1 project built upon the code or requirements of a previous project; however, none of these utilized peer review.

Although we understand that peer review may be utilized in more visualization courses than we identified in our survey, the results suggest a need for expanded evaluation and discussion of the benefits of peer review in the visualization classroom. The only peer review evaluation that we are aware of is our work from the 2017 Pedagogy of Data Visualization IEEE VIS workshop, where we presented a rubric for peer review for visualization courses [14]. The rubric included 5 main categories of mix-and-match style evaluation criteria that can be customized to a wide variety of contexts. This work is built around that rubric and the formalization of critique within a visualization course through peer review.

3   Course Overview

With the extensive usage of peer review in other disciplines, we endeavored to build a visualization course with peer review as a core component. Our course, titled "Data Visualization", was taught Spring semester of 2017, 2018, and 2019 and was a co-listed undergraduate and graduate course. The course was hosted in the Computer Science department, within the College of Engineering at the University of South Florida, and the educational emphasis of the course was primarily good visualization practice, though a strong emphasis was placed on software design as well.

3.1   Course Content

The content of the course was primarily taught following Munzner's *Visualization Analysis & Design* [33] and the Nested Model [32] with additional outside visualization content (e.g., Vis Lies[4],

---

[3]We do not include "peer evaluation" of ones own group as peer review.

[4]Vis Lies: `http://www.vislies.org`

New York Times Graphics[5], etc.) and research papers added throughout the semester. We decided to pursue structured assessment (i.e., well-defined projects), leaving Problem Characterization aspects of the Nested Model to in-class presentation, activities, and discussions. The remaining levels, Data Abstraction, Encoding and Interaction, and Algorithm Design, were covered in class and emphasized through the projects. The presentation methods were primarily lecturing, research paper presentations by students, and discussions (e.g., small group and whole class critiques).

The course learning objectives (see syllabus in supplementary materials) were that students would demonstrate the ability to:

[L1] Build effective visualizations by evaluating a provided data and user requirements and pro-gramming an interface to match those requirements.

[L2] Associate visualizations with the foundational components, e.g., data abstractions and visual encodings, that go into their construction.

[L3] Critique the effectiveness of interactive visualizations with respect to task selection, visual encoding choices, and interaction design and implementation.

To satisfy [**L1**] and [**L2**], the course consisted of 8 projects (see section 3), 1 in Tableau[6] and 7 in Processing[7], totaling 50% of their final grade. Projects were due every 10-14 days, except for Project 6 that allowed approximately 30 days since it was assigned over Spring Break. To satisfy [**L3**], peer review of projects was integrated into the course. Upon the completion of each project, students were asked to provide reviews of 3 randomly selected peers' work using a provided rubric within 5-7 days (see section 5). The peer feedback served as the primary form of qualitative feedback students received. In addition, the Spring 2018 version of the class included self-review after the completion of peer review. Both peer review and self-review were for small amounts of (spot checked) *completion-based* credit (approximately 10% of the final course grade). The instructor and teaching assistants still completed project grading. Grades were assigned based primarily upon objective requirements, as well as some subjective judgment.

3.2 Evaluation Methodology

There are significant subjective measurements necessary to evaluate the effectiveness of a visualization course [52]. Visualization education research also lacks a standard rubric for measuring engagement. In a review of 5 major visualization venues (InfoVis, SciVis, VAST, EuroVis, and Pacific Vis), we found no form of empirical evaluation of students' work or engagement in the classroom. Thus, we have conducted a qualitative evaluation using questionnaires to measure *perceived* improvement and engagement. We supplement and reinforce the analysis by combining the questionnaire data with both qualitative analysis using representative examples of student work and quantitative analysis through natural language processing.

A useful reference for our non-experimental evaluation methodology can be found in the field of writing composition. For example, Mulder et al. used similar qualitative and quantitative methodologies, based on content analysis from peer review comments and student questionnaires [31]. Furthermore, analyzing the number and variety of words used in written comments has been used for measuring both learning outcomes [26] and student engagement [51].

---

[5]New York Times: https://www.nytimes.com
[6]Tableau: https://www.tableau.com/
[7]Processing: https://processing.org/

3.3   Data Collection

The student-produced data falls into the following categories:

Student Visualizations   Project submissions were gathered from 8 projects each year (2017-2019). We manually reviewed projects to find those which, in combination with peer comments, exemplify the value of peer review. We have included 1 such sample project in the supplemental materials[8].

Student Peer Review and Self-Review Comments   Student peer review comments were collected on 8 projects from 2017 and 7 projects from 2018 and 2019. Student self-review comments were collected on 7 projects from 2018 only[9]. We analyzed them with a dictionary-based natural language processing algorithm [3] for matching positive and negative keywords to produce numerical feedback including overall sentiment (positive or negative) of the text, counts of parts of speech (i.e., noun, adjective, adverb), the average length of comments, etc. The algorithm includes an *aspect extractor* (similar to one developed by Google towards analyzing reviews of local services, such as restaurants and hotels [7]) that scans text in a sliding window and produces a list of important aspects (i.e., nouns), which are in close proximity to sentiment words (i.e., adjectives). The idea is that these aspects are frequently commented upon words associated with either positive or negative sentiment, indicating their importance to reviewers. Figure 2 shows an example adjective ("colorful") and noun ("images") match within the sliding window, highlighted in teal. While "colorful" has inferred positive sentiment, it is modified by a comparative adjective, "more", which reverses its sentiment to negative. It is notable that this algorithm was developed for peer review in engineering courses, but not explicitly tuned for visualization courses or the particular rubric utilized. Thus, the algorithm does not "look" for visualization keywords, only for general sentiment-producing words.

Post-Course Questionnaire   Each year (2017-2019), students were asked to complete an optional 20-question post-course questionnaire, given at the conclusion of their final exam and before receiving their final grade. Students were offered a small amount of extra credit to anonymously answer questions ranging from demographic information (see Figure 3) and anticipated final grade to helpfulness of review comments and open-ended suggestions for improving the peer review process. Numerical answers were all placed on a 5-point Likert scale. The overall participation rate was 98%. We performed quantitative analysis on numerical answers and manually reviewed written feedback for qualitative results. The questionnaires and student responses are included in our supplementary materials.

---

[8]Due to the sheer number of projects (over 1000), we were unable to anonymize all projects for inclusion in the supplemental materials.

[9]Self-review was dropped in 2019 due to concerns about workload.

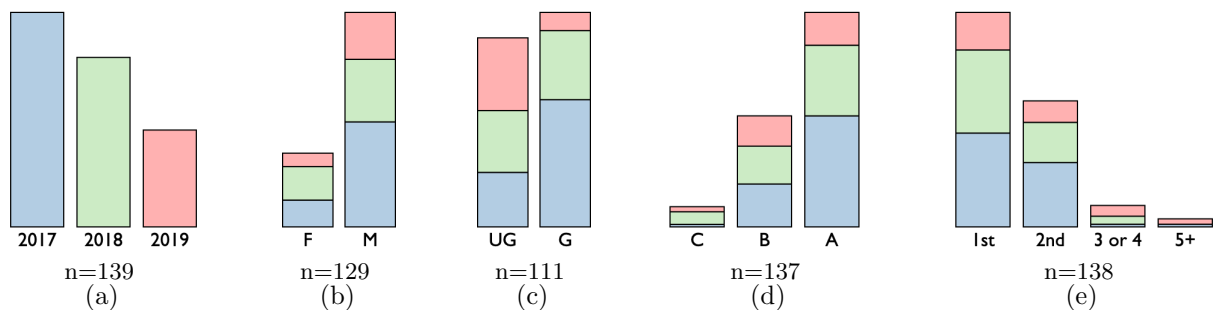Figure 2: Aspect Extractor: noun in close proximity to an adjective.

Figure 3: Histograms of questionnaire respondent demographics, including (a) students per year, (b) gender, (c) undergraduate vs. graduate, (d) anticipated course grade, (e) times using peer review in a CS course. Color-code: ▢ 2017 / ▢ 2018 / ▢ 2019

4 Project Design

When designing projects, our interest was to see students gain visualization skills (i.e., [**L2**]) by demonstrating proficiency in using software engineering problem-solving techniques (i.e., [**L1**]) [8]. To support this dual goal of visualization practice and software design, the course projects were divided into 4 categories of emphasis:

▢ Familiarization and Expectation Setting: This project familiarizes students with using standard visualization types.

▢ Visualization Foundation Building: These projects emphasize the implementation of basic mechanics of visualization, such as data abstraction, visual encoding, and interaction.

▢ Applying Skills to New Contexts: In these projects, students apply their skills to develop more complex visualizations.

▢ Software System Engineering: These projects use software engineering skills to build interactive multi-view interfaces.

The peer reviews serve as an integral part of the future projects by allowing students to use feedback to refine the work they submit (i.e., [**L1**]). Therefore, the projects are set up to maximally build upon and reuse components from previous projects, while still challenging students with new project requirements. Figure 4 shows the timeline of projects over the semester. The solid black edges indicate direct reuse of components and associated feedback from previous projects. Full project descriptions are included in our supplementary material.

4.1 Project Descriptions

▢ Project 1: Building a Visualization Story
Objective: The goal of the project, which was assigned the first day of class, is for students to become familiar with a fully functional visualization system and to get them thinking about objective/task-based visualization design. *Requirements:* Using Tableau, the students are to: import a dataset of their choice, develop 3 "questions" about the data, choose 3 or more visualizations to answer those questions, and finally place the results into a short narrative.

▢ Project 2: Familiarization with Processing
Objective: This project provides students a first glimpse at the Processing development environment by developing simple interfaces with simple data. *Requirements:* The students are asked to create

3 sketches: a bar chart, a line chart, and a combined bar and line chart. Although the students have had little exposure in class, they are encouraged to use their best judgment concerning coloring, scales, labeling, tick marks, and other embellishments.

▢ Project 3: Reusing Drawing Objects

Objective: The project provides a lesson in building reusable interfaces. *Requirements:* The students are to build 2 sketches: first, a scatterplot, and second, a scatterplot matrix (reusing the scatterplot object) with a dataset of approximately 270 data items and 4 attributes. The students are encouraged to use additional visual channels, wherever possible, and again to use their best judgment concerning scales, labeling, tick marks, and other embellishments.

▢ Project 4: Adding Interaction

Objective: In this project, students are encouraged to explore options for improving their visualizations using interaction. *Requirements:* Students are required to add interactions to their sketches from Projects 2 and 3. For the line chart, bar chart, and scatterplot, this is in the form of data item selection. For the scatterplot matrix, the selection provides the attributes used in a detail view scatterplot. *Utilization of Feedback:* This project provides the first opportunity for students to directly apply the feedback they have received. They are actively encouraged to use the feedback from Projects 2 and 3 to improve the aesthetics and functionality of their visualizations.

▢ Project 5: Advanced Visualization Interface

Objective: This project asks students to use the knowledge they have built in the previous projects to build an advanced visualization interface—parallel coordinates plot—from scratch. *Requirements:* Build a parallel coordinates plot with interactions, such as axis swapping and brushing.

▢ Project 6: Building a Dashboard

Objective: The objective of this project is to combine previous projects into a single linked-view dashboard interface. *Requirements:* Create a dashboard using the bar chart, line chart, scatterplot, scatterplot matrix, and parallel coordinates plot from Projects 2-5. Provide linked-view interaction via selection. Students are encouraged to add additional interactions and to use their best judgment concerning dashboard layout. *Utilization of Feedback:* Since this project combines the work of Projects 2-5, students are encouraged to revisit their previous designs and associated feedback to improve the look and functionality of their dashboards.

▢ Project 7: Adding Aggregation

Objective: The objective of this project is to extend the dashboard with capabilities that better support large data through statistically derived attributes. *Requirements:* Extend the dashboard from Project 6 to include histograms and corrgrams (using both Pearson Correlation Coefficient and Spearman Rank Correlation Coefficient) of the data. *Utilization of Feedback:* The students are encouraged to use their feedback from Project 6 (in addition to Projects 2-5) to improve the design of their dashboard.

▢ Project 8: Drawing Graphs

Objective: This project is mostly independent of the others, enabling students to translate lessons learned to a new context. *Requirements:* Visualize a graph using a force-directed layout. In addition, the students must provide the ability to interact with the layout.

5 Peer Review

Upon the completion of each project, students were asked to provide reviews of 3 randomly selected peers' work within 5-7 days. One point of variation between the semesters was the platform used for code submission and subsequent peer review. For project submissions, we experimented with zip files via Canvas assignments (2017), Bitbucket[10] (2018), and GitHub Classroom[11] (2019).

Peer review in 2017 was handled via Canvas' built-in peer review platform, which provides a liberal arts style of peer review, where a document is displayed, and questions appear alongside. In our case, students had to download and run code on their local machine. Once feedback was submitted, it would immediately become available to the recipient. In 2018, peer reviews were assigned and submitted via a Canvas quiz. At the end of the peer review period for a project, the

---

[10]Bitbucket: `https://bitbucket.org/`
[11]GitHub Classroom: `https://classroom.github.com`

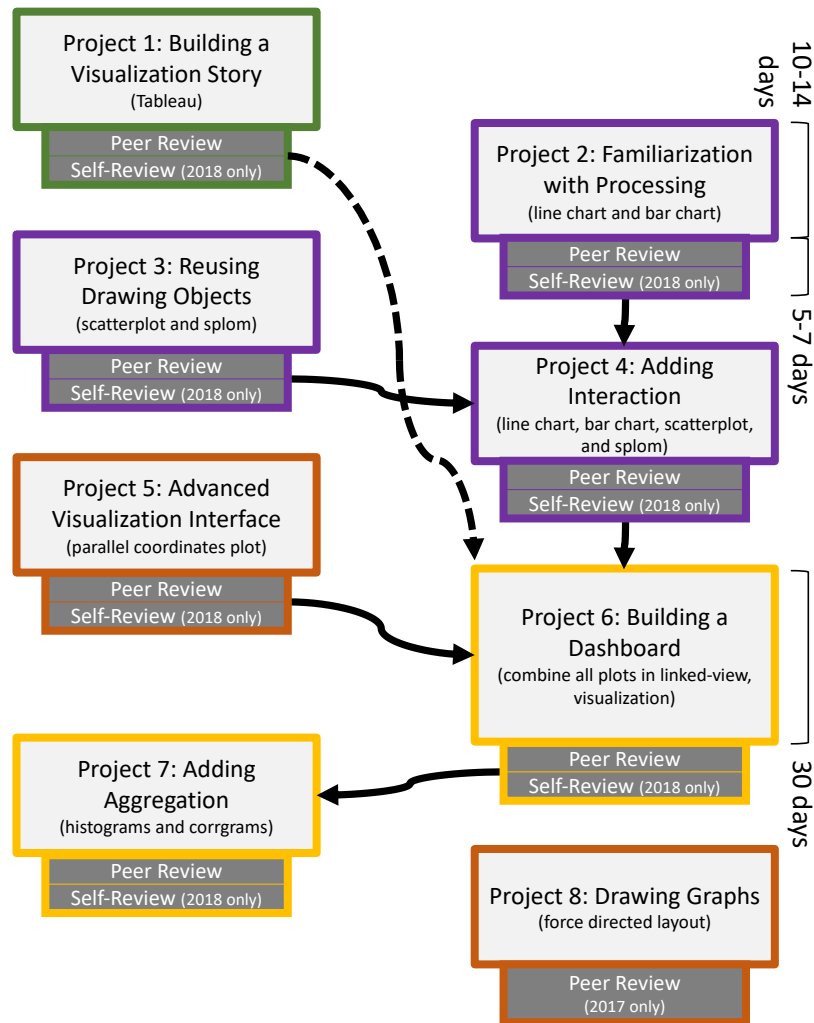Figure 4: A timeline of course projects and student-driven feedback. The — solid edges are directly utilized feedback in future projects, while - - - dashed edges are indirect utilization. Projects are in 4 broad categories of objectives, including: familiarization; foundation building; applying skills in new contexts; and software engineering.

feedback was returned by e-mail using custom scripts. In 2019, Google Forms was used to capturing feedback and delivered via a custom webpage at the end of the peer review period.

Figure 5 shows the post-course survey response to the question of whether the interface "worked well". These are independent samples, with no common baseline, making direct comparison impossible. However, the students using the Canvas peer review system had the most favorable view of that platform (see Figure 5a), followed by the Goggle Forms group (see Figure 5c), then the Canvas Quiz group (see Figure 5b). In the free-response section of the survey, several students in the Google Forms group stated that better integration with Canvas would have improved the experience.

5.1 Peer Review Rubric

A rubric is a ubiquitous pedagogical tool that articulates the expectations for an assignment by listing the criteria and describing levels of quality from excellent to poor. They are used by a large number of instructors in a variety of disciplines to provide feedback on and to grade an array of student products, e.g., writings, oral presentations, portfolios, projects, etc. [25, 53, 13].

A Rubric for Visualization Education We addressed the limited availability of rubrics for peer review in the visualization classroom with our rubric at the Pedagogy of Visualization 2017 workshop [14]. We have included a copy of the rubric in our supplementary materials, and a LATEXversion of the rubric can be downloaded/cloned/forked at `https://github.com/USFDataVisualization/VisPeerReview`. We briefly revisit the structure of the rubric.

The rubric was built by carefully reviewing the course content and extracting key concepts necessary for demonstrating proficiency in learning objectives [**L2**] and [**L3**]. The basic structure of the rubric divides topics into 5 major assessment categories, with each category having 3 sub-assessments affixed to a 5-point scale. Each sub-assessment contains a comment box for details on the scoring.

The 5 major assessment categories are: algorithm design, interaction design, visual design, design consideration, and visualization narrative. The algorithm design category is concerned with algorithm selection and implementation. Interaction design is concerned with how the user interacts with the visualization. Visual design is concerned with the technical aspects of how data are placed in the visualization (e.g., visual encoding channels, their expressiveness, and their effectiveness). Design consideration focuses on the composition and aesthetic aspects of the visualization, such as embellishments. The final category, visualization narrative, is used in projects where the story surrounding the visualization is as important as the visualization itself.
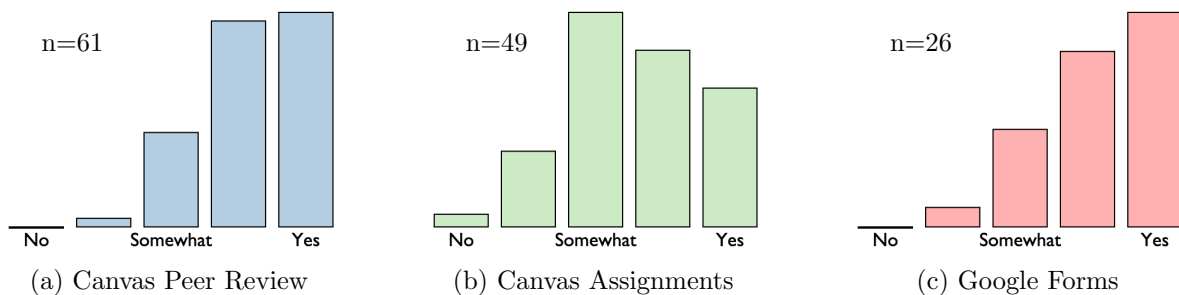

(a) Canvas Peer Review   (b) Canvas Assignments   (c) Google Forms

Figure 5: Post-course survey results for "how well" the peer review interfaces worked. Color-code: ■ 2017 / ■ 2018 / ■ 2019
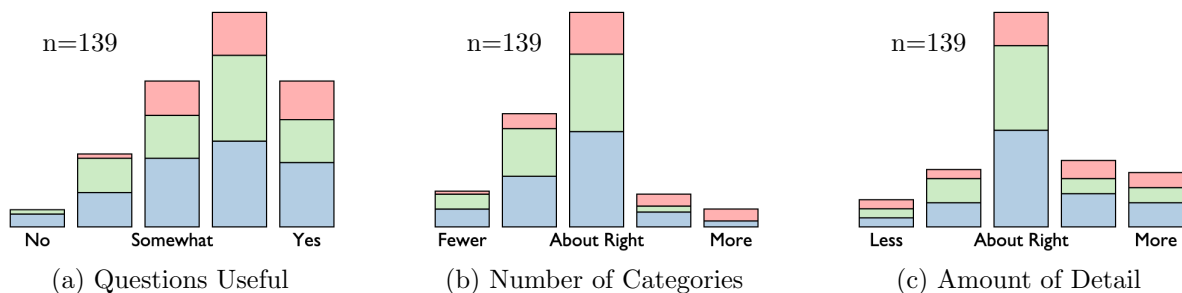
Figure 6: Questionnaire results for questions about the rubric. Color-code: [ ] 2017 / [ ] 2018 / [ ] 2019

Rubric Customization In the original design of the rubric, we intended a certain level of customization to be applied based upon the content of an assignment or course. For each project, we extracted the relevant components from the full template. Project 1, for example, included a narrative component, while no other project included such a requirement. Projects 4-8 had interaction components, while Projects 1-3 did not. Furthermore, the sub-assessments included in the early project rubrics reflected topics that had been covered in class. The rubrics for all projects can be found in the supplementary materials. Furthermore, an example can be seen in Figure 1 (right) for the assessment received from 4 peers to the submission in Figure 1 (left). The example shows 3 main categories, visual design, design consideration, and interaction, along with 16 sub-assessment questions for Project 6.

Rubric Evaluation As instructors, the rubric covers all of the key concepts we intend for students to master. On the post-course questionnaire, we asked the students 3 questions related to their opinions of the rubric (see Figure 6). Their opinions indicated that, while the rubric questions were useful (see Figure 6a), fewer (see Figure 6b) with the same level of detail (see Figure 6c) would be preferred. We are currently considering methodologies for reducing the number of questions, e.g., combining similar sub-assessments over the course of the semester.

6   Peer Review and Student Learning Outcomes

While assessment is the heart of formal higher education and a core component of effective learning [35], for this work, we do not prioritize evaluating the correctness of a visualization, i.e., how it corresponds to a professor's grade. Rather, we are interested in evaluating the *influence* peer review had on students producing the visualizations.

Specifically, we limit our analysis to 3 questions:

1. Does peer review reinforce course content?

2. Do students engage in and enjoy the peer review process?

3. What aspect of peer review is most beneficial to students?

6.1   Does Peer Review Reinforce Course Content?

We primarily utilized the student peer review comments to determine whether peer review reinforces course content. If students mention key concepts learned in the course in written responses, we

interpret it to mean that they took the opportunity to identify course content in context (i.e., learning objective [**L2**]).

The open-ended review comments (peer=3104, self=339, total=3443) were gathered from multiple comment sections on each review form. Each comment section was concatenated into a single string and analyzed by the process described in subsection 3.3.

We first compared whether students were using terminology from the rubric or whether they were commenting on other things. Included in the top 20 aspects are the following words: visualization, legend, color, ink, data, type (of data), graph, information, ratio (of data), use (of color, encoding), scale, amount (of ink, data), interaction, chart, lie, and density. These words are all found on the rubric, so students are likely parroting the terminology. Nevertheless, we posit that this forces the students to identify course content in the context of their peers' work. Figure 7 provides some context for the utilization by displaying the aspects in the center column with the number of connected positive sentiment words (left column) and the number of connected negative sentiment words (right column).
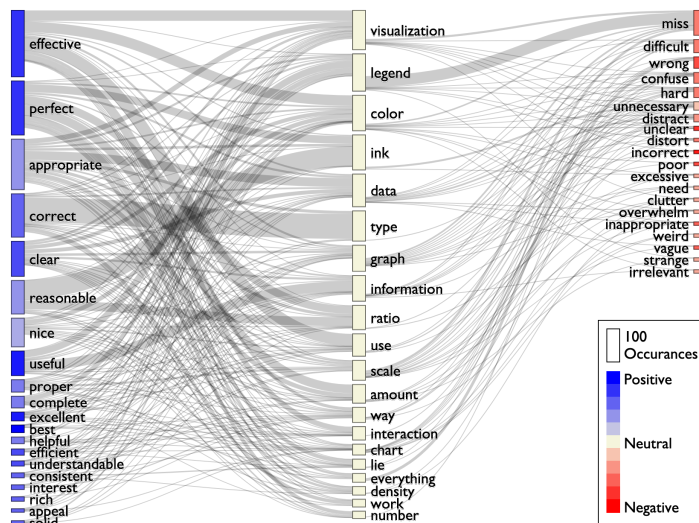

Figure 7: Visualization of the sentiment of peer reviews. Aspects (i.e., nouns) appear in the center column connected to positive (left) and negative (right) sentiment words that they appear by in text. The height of the bars indicates the number of occurrences. Color saturation indicates how positive or negative the sentiment is.

Although we cannot directly measure whether students understand the concepts better through peer review, we can be confident that this approach provides repeated exposure to key concepts (giving and receiving on-topic feedback), and students are, at the very least, repeating terminology.

Finally, comments from the post-course questionnaire like the following seem to confirm that students note course content in each others' visualizations:

> "I'd definitely recommend keeping it . . . It also helps those struggling with concepts to see how others did it, to do it better on future assignments."

> "Peer review helps in understanding data visualization principles . . . This helps a lot in doing future assignments and understanding my mistakes."

6.2   Do Students Engage In & Enjoy Peer Review?

To quantitatively evaluate student engagement, we analyze the number and variety of words written as recommended by [51]. We specifically tag nouns (aspects), adjectives (aspect modifiers), and adverbs (sentiment enhancers) from the peer review comments. There are a variety of other sentiment metrics provided by our algorithm (e.g., sentiment score, percent of reviews that lacked sufficient information to score, the average purity of positive sentiment) that make little sense unless comparing between individual projects. The relevant summary statistics are shown in Table 1.

We noticed that undergraduates wrote more than graduate students, with more words per sentence and a greater variety of tagged parts of speech (especially adverbs). Interestingly, the

Table 1: Analytic evaluation of peer comments

	Reviews	Avg Keywords	Avg Positive Keywords	Avg Negative Keywords
Graduate	1543	7.75	4.99	2.76
Undergraduate	1573	9.26	5.78	3.48

	Avg Words Per Review	Avg Words Per Sentence	Avg Adjectives	Avg Adverbs	Avg Nouns
G	100.31	7.85	9.78	4.91	30.50
U	135.61	10.45	12.42	8.37	36.86

ratio of negative to total keywords was similar for undergraduates (38%) and graduates (36%), which is an important measure of engagement because critically evaluating a visualization requires more investment than just a cursory review—it requires applying learned concepts to explain *why* something is wrong (e.g., analyzing for "lie factor"). Considering the length, variety of parts of speech, and the ratio of negative keywords may indicate that undergraduate students are slightly more invested in the peer review process than their graduate peers. Many post-course questionnaire comments reflected a general sense of increased engagement and motivation:

> *"Process is interactive and healthy."*

> *"I enjoyed it! Saw some really good work by my peers that motivated me in the final projects."*

> *"Definitely worth the time and energy, learned a lot by examining code, which helped to see their thought process."*

To discover whether students enjoyed the peer review process, we asked 2 questions on the post-course questionnaires (see Figure 8): (1) if students believe they learned more because of the peer review process (perceived improvement); and (2) if students recommend continuing peer review. 82% of respondents reported learning at least somewhat more (score of 3 or more; mean = 3.6). To validate consistency across semesters, we ran unpaired t-tests comparing 2017/2018: $t(107)=0.32$, $p=0.75$ and 2018/2019: $t(73)=1.6$, $p=0.11$, which showed no statistical difference (i.e., $p > 0.5$).

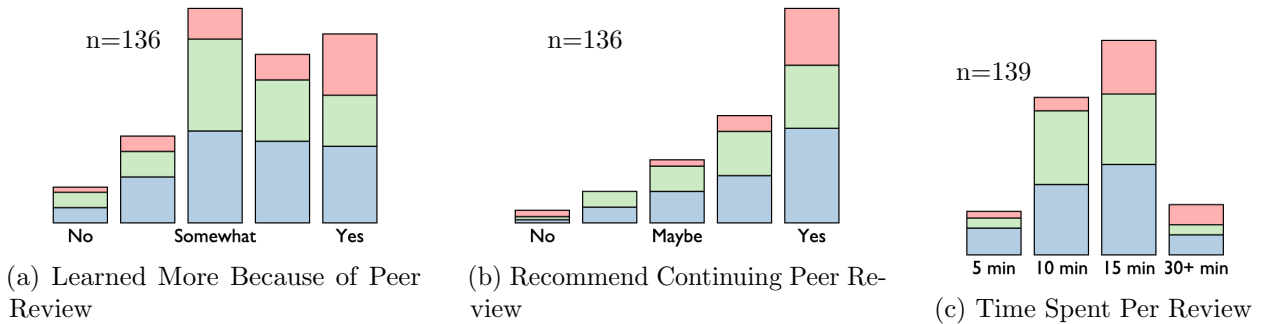Additionally, 75% recommended continuing the process (score of 4 or more; mean = 4.1, 2017/2018:

(a) Learned More Because of Peer Review

(b) Recommend Continuing Peer Review

(c) Time Spent Per Review

Figure 8: Questionnaire results related to (a-b) engagement and (c) effort. Color-code: ☐ 2017 / ☐ 2018 / ☐ 2019

$t(107)=0.65$, $p=0.51$ and 2018/2019: $t(73)=1.5$, $p=0.15$) with almost half of students strongly recommending it, despite the fact that reviews were more work, taking between 10 and 15 minutes to fill out (see Figure 8c). Finally, some students enjoyed the process so much that they recommended increasing the stakes of peer review in the course:

> "I liked being able to view what others did because it helped me see ways I can improve my work. I also think the peer reviews should be worth more points."

> "I think you should showcase the best and worst visualizations in class. :D Let the students praise/rip apart them as an exercise. Its also more motivation to do well."

6.3 What Aspect Is Most Beneficial to Students?

To determine which aspect of peer review is most beneficial to students, we asked 3 questions about the helpfulness of content: receiving feedback (Figure 9a), providing feedback to others (Figure 9b), and participating in self-review (Figure 9c). Interestingly, in order from most helpful to least, most students found (1) reviewing others' work (mean = 3.9, 2017/2018: $t(109)=0.71$, $p=0.42$ and 2018/2019: $t(75)=1.1$, $p=0.28$); then (2) self-review (mean = 3.4); and finally (3) feedback received (mean = 3.2, 2017/2018: $t(109)=0.20$, $p=0.74$ and 2018/2019: $t(75)=0.56$, $p=0.58$) helpful. Thus, it appears that students perceive the maximum benefit from their ability to review others' work (learning objectives [**L2**] and [**L3**]), not by receiving feedback on their own, which is consistent with Garousi's findings [15] (see subsection 2.1). Students leaned towards preferring textual (as opposed to numeric) responses (see Figure 9d), but a surprisingly low rating was received for the quality of peer reviews (mean = 3.2, 2017/2018: $t(108)=0.08$, $p=0.99$ and 2018/2019: $t(74)=1.7$, $p=0.10$) (Figure 9e) which may explain why nearly 25% of students never or rarely looked at their feedback (Figure 9f).

Finally, many students mentioned explicitly in the course questionnaire how helpful it was to view others' code and visualizations:
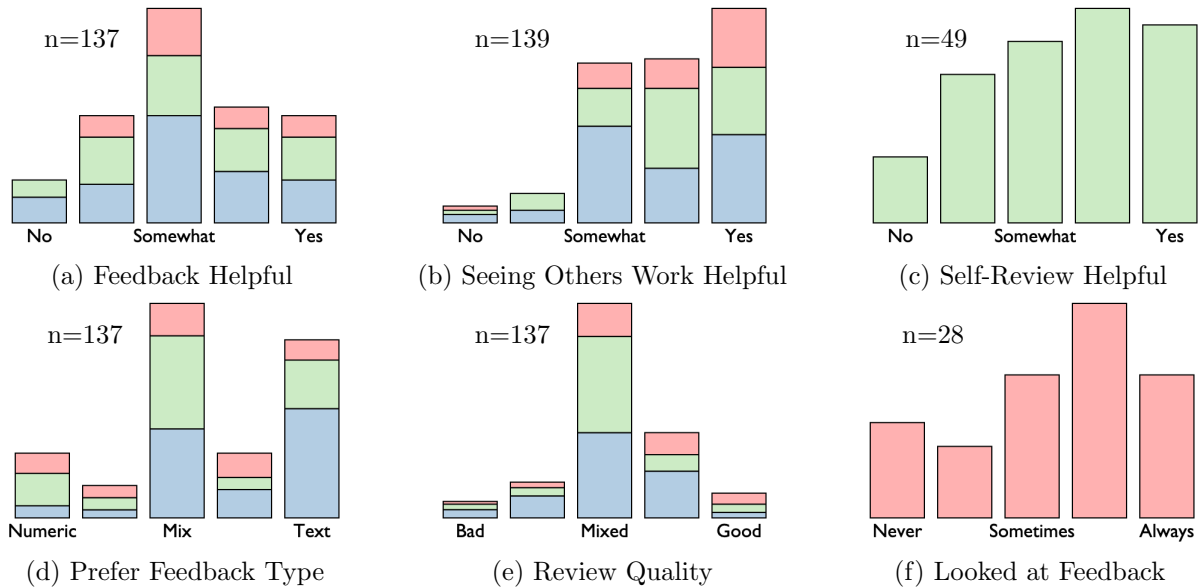

(a) Feedback Helpful

(b) Seeing Others Work Helpful

(c) Self-Review Helpful

(d) Prefer Feedback Type

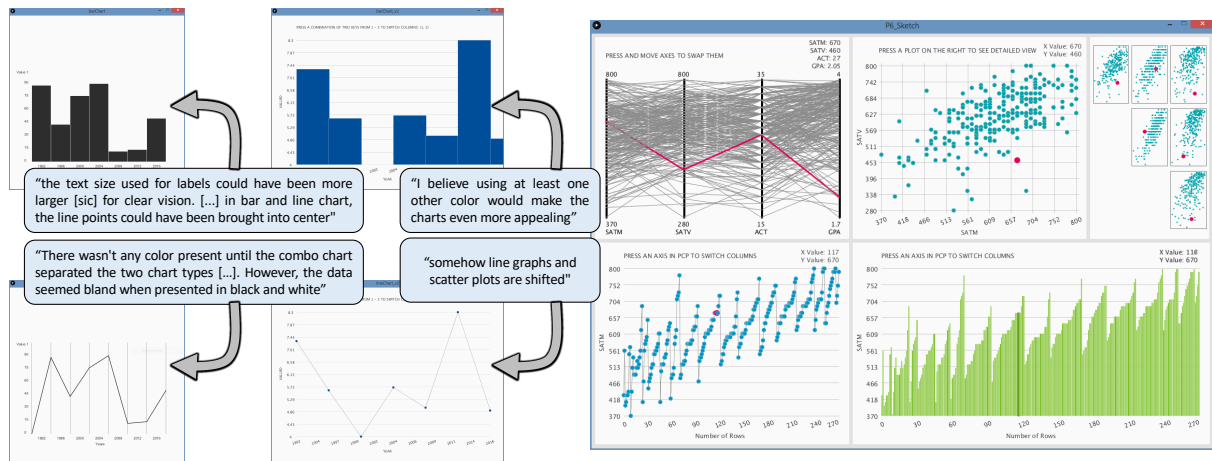(e) Review Quality

(f) Looked at Feedback

Figure 9: Questionnaire results related to the perceived utility of feedback. Color-code: ▨ 2017 / ▨ 2018 / ▨ 2019

14

(a) Bar & Line Chart from Project 2 "Familiarization with Processing"

(b) Bar & Line Chart from Project 4 "Adding Interaction"

(c) Dashboard from Project 6 "Building a Dashboard"

Figure 10: Example series of 3 projects and associated feedback utilized to improve the design.

"Looking at other peoples code helped. Grad students = bad code."

"I think it's a good idea; it helped me by giving examples of what not to do mostly."

"I am very impressed with this process. It is very helpful to students."

"The way its done gives you a chance to learn from many different peers as well as help teach many."

6.4 Peer Review Examples

Although we did not control for peer review, the following examples of particular students' progress throughout the semester (learning objective [**L1**]) reflect a combination of instruction and the effectiveness of the peer review process.

In the first example, we highlight differences between Projects 2, 4, and 6 to demonstrate the effect of a student receiving and implementing peer feedback. Figure 10a received the following comment(s): "the text size used for labels could have been more larger [sic] for clear vision. [...] in bar and line chart, the line points could have been brought into center" and "There wasn't any color present until the combo chart separated the two chart types with a red color. However, the data seemed bland when presented in black and white." In response, the student changed the color of the bar chart, moved the line chart points to the center of the label (actually displaying the points themselves) and made the axis titles slightly larger, as shown in Figure 10b. Figure 10b received the following comment(s): "somehow line graphs and scatter plots are shifted" and "I believe using at least one other color would make the charts even more appealing". In response, the student added another color to Figure 10c and shifted the appropriate graphs to not overlap. Thus, in addition to benefiting from reviewing others' work, the student appeared to consider and implement much of the feedback they received.

In some cases, even if a student received no useful feedback, we still noticed continuous improvement, as in Figure 11. Clearly, something was influencing the improvement, and one could speculate that seeing others' work contributed to this. However, we have no way to confirm this hypothesis.
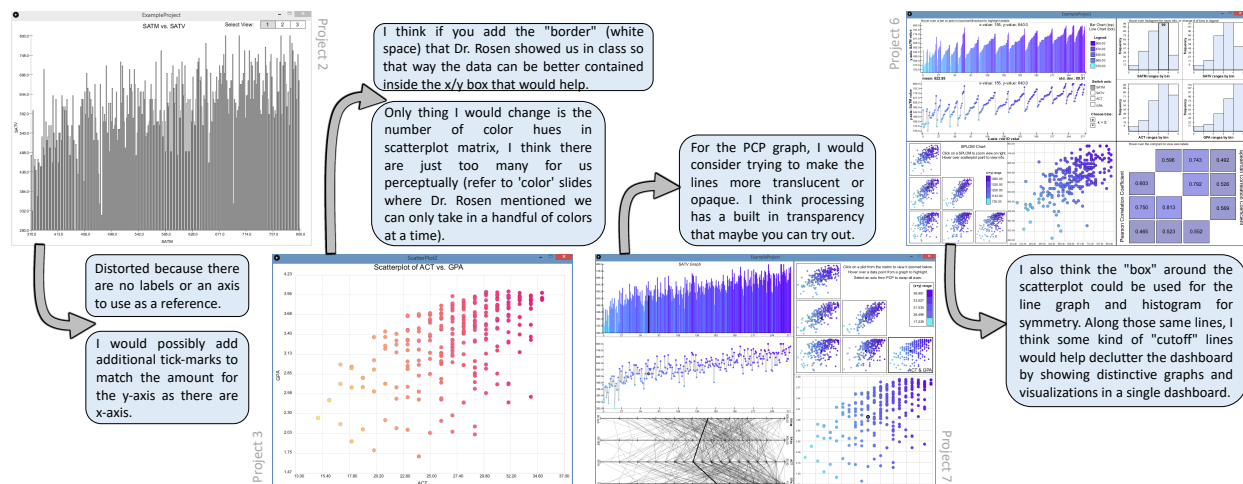
Figure 11: Illustrative example of a student's projects and the feedback they *gave* to their peers, reflecting applied concepts.

Another interesting observation from reviewing feedback was that students tend to comment on others' work in avenues that they have already implemented in their visualizations (showing progress in learning objectives [**L2**] and [**L3**]). Figure 11 shows student projects and the feedback the student *gave* to their peers. For example, in Project 2, the student mentions axis ticks and labels—something they carefully implemented in their bar and line charts. In the Project 3 feedback, the student points out the correct use of colors and directly references materials learned in class. The student references a specific programming technique for avoiding clutter in a PCP in their Project 6 feedback, and finally, a tip to delineate charts on the dashboard for Project 7 feedback. In each situation, the student offers advice that corresponds to a technique they correctly implemented, which corroborates our findings in subsection 6.1 that peer review reinforces course content by allowing students to communicate recently learned and applied concepts during the peer review process (an opportunity they might not otherwise have).

6.5   Instructor Perspective: Observed Student Benefits

The previous examples have shown, from student data, how peer review benefits visualization students. We have observed several additional student benefits that we informally evaluate.

Formalizing Existing Collaborations   The teaching assistant and grader for the 2018 and 2019 data visualization course pointed out one advantage of peer review related to engagement, not previously discussed. Many, but not all, students will discuss projects with one another and seek feedback without instructor intervention. He noted, however, that peer review using a rubric forces (in a positive way) interaction for some students who would not otherwise interact with and offer constructive criticism to others. To some, it offers a more comfortable environment than discussion in class. In all cases, it provides structure to the feedback. Requiring feedback in this way is important to the student (to practice critical evaluation skills), as well as to the visualization (a greater variety of perspectives contribute to increased effectiveness).

Increasing Ownership   Multiple students commented on the value of seeing each others' work. There are 2 effects that we observed: (1) it enables seeing how peers have solved a problem in order

to redesign your visualizations; and (2) there is pressure to perform better when you know your peers will see your work. From our conversations with students, we felt that this gave the students more ownership in the projects, many discussing with pride the final product they developed.

Critical Evaluation Skills   Critical evaluation of others' work is a mandatory skill for academia, for assessing research quality, and in industry, for code and design reviews. Many engineering programs are knowledge and skills-based (ABET accreditation criteria reinforce this) while teaching critical evaluation informally. Peer review can help fill this gap in developing skills.

7   Implementation in Other Courses

This paper intends to encourage other educators to adopt and customize the methodology for their courses. By-and-large, the approach we have used is not restricted to the style of computer science visualization course we taught but is highly generalizable to other design-oriented or non-computer science visualization courses. Most components can be swapped out—e.g., the projects can be swapped for other projects, as long as they emphasize iterative design; the rubric can be replaced by other evaluation criteria, e.g., Design Activity Framework worksheets [27]; etc.

7.1   Implementation Options

We provide our course and recommendations as a road-map for other instructors with 4 routes to implement peer review in other classrooms:

Simple: Use our project and rubric designs. This approach is less than ideal for most since each instructor has their own perspective on the most important concepts and skills to test.

Lightweight Integration: Peer review can be added to existing projects using our rubric or a rubric developed by the instructor. This approach lacks the feedback cycle but will still capture some of the peer review benefits, such as reinforcing course concepts.

Middleweight Integration: If the course already uses a continuous design process (e.g., a semester-long design project), incorporate one or more peer review stages using our rubric or a rubric developed by the instructor. In addition to reinforcing concepts, this integration enables using feedback for a continuous-improvement process.

Heavyweight Integration: The course projects can be completely redesigned, such that they are built around peer review (instead of the other way). Projects would be specifically designed to emphasize course topics, with each project cycle building on the previous. This approach would capture all of the benefits of peer review.

Platforms for Delivery   The platform for delivering content can have a significant impact on student perception of that content. Of the 3 platforms we used (see section 5), each had advantages and disadvantages, but in the end, we found none of them ideal. Most likely, an instructor adding peer review to their course would choose the peer review mechanism already available on their LMS.

Instructor Effort   On the surface, peer review appears to potentially result in substantial time savings for instructors no longer needing to give subjective feedback. For example, 60 students, 8 assignments per semester, and 10 minutes of feedback per assignment should result in over 80 hours saved. However, much of that time is not saved but actually diverted to other activities. On one hand, additional administrative tasks arise—grades still need to be assigned, peer review requires

additional effort to set up and administer, and peer reviews need to be monitored for quality. At the same time, there is an opportunity and need to provide more detailed instructor feedback to those highly engaged students who request it—we regularly advertised that instructor feedback was available, in person, upon request (e.g., during office hours). Peer review is a complement to existing educational approaches, but the instructor, as the expert, still needs to be involved.

Completion and Quality   Incentives in peer review have been widely studied in the context of writing courses [16, 38, 49, 50]. We achieved a high completion rate, 95%, by making peer reviews a mandatory part of the course, spot-checked, and graded on completion. Unfortunately, the quality of the peer reviews students received varied (see Figure 9e). Just over 56% of the students rated the quality of reviews they received as "Mixed", with only 6% stating that the quality was "Always Good". Increasing the stakes (i.e., making peer review quality a significant of the grade received) would likely improve quality, but create substantial overhead for instructors, as detailed analysis of every peer review submitted would be required.

7.2   Risks

When considering using peer review, the risk for collusion, malice, and cheating is great, even with double-blind reviews.

Peer Review for Grading   Instructors should take care to randomize peer reviewers, and grades should (at most) only be loosely based upon the results of the peer review. However, it is also important to remember that there is both art and science to design with no single optimal design, making the wisdom of the crowd potentially useful in assessment. Building machine learning models that utilize peer evaluation to (semi-)automatically assign grades is an exciting direction for future study.

Collusion   If reviews are used for grading, the problem of collusion or malice may play a role (e.g., the review of a friend may be overly optimistic). Keeping all submissions anonymous is helpful, but anonymity is hard to maintain, especially in a smaller educational setting where students may talk and discover they are evaluating each others' visualizations.

Cheating   Providing access to other students' projects is an obvious risk. In fact, we have dealt with 2 such cases of students stealing others' code through peer review. However, frequent and repeated warnings, along with detection systems, e.g., MOSS (Measure Of Software Similarity) [44], can reduce the number of such incidences.

Falling Behind   Since many projects build on one another, students who fail to complete early projects to a high standard may continue to struggle on future projects in a snowball-like effect. We mitigate this by providing a significant amount of time, almost 1 month for the major integration project (Project 6). Nevertheless, the risk of frustrating and losing students early remains.

8   Conclusion

We have presented our experience and evaluation of peer review in the visualization classroom. This approach has 2 significant benefits. First, it provides a framework for engaging students through

critical evaluation of visualizations. Second, it is a mechanism for providing students with diverse and timely feedback on their work.

Several implementation issues deserve further study. We utilized a variety of peer review interfaces, all of which had inflexible designs, not well suited to visualization peer review. Having a tool that enables us to describe the grammar of a graphic concisely would help to gain insights into the structure that underlie statistical and programming languages to produce better graphics [60, 59].

The rubric itself has a focus on low-level details of design (e.g., visual encodings, tick-marks, and labels), as opposed to high-level design, such as composition and choice of technique. This focus is partially an artifact of our project design. We do not believe that the rubric, as presented, will be the final static version. We anticipate it will be a growing and evolving document as community members provide their input and the focus of the visualization community changes. Nevertheless, further evaluation of potential rubric designs should be considered.

A final direction is the customization of the rubric through reducing constraints. Assuming the critical thinking skills of students are weak, particularly in the domain of visualization, the rubric itself can be a tool to help improve those skills. At the beginning of the course, the rubric can include all scoring categories, and as the course progresses, categories can be combined and removed. In this way, students will go from highly structured to free-form evaluation.

Disclosure   The first author (Beasley) was a student in the 2017 graduate course. His participation in the project began *after* the course was completed.

Acknowledgments

References

[1] A. Bacchelli and C. Bird. Expectations, outcomes, and challenges of modern code review. In *Intl. Conf. on Soft. Eng.*, pages 712–721, 2013.

[2] Z. J. Beasley, L. A. Piegl, and P. Rosen. Ten challenges in CAD cyber education. *Computer-Aided Design and App.*, 15(3):432–442, 2018.

[3] Z. J. Beasley, L. A. Piegl, and P. Rosen. Designing intelligent review forms for peer assessment: A data-driven approach. In *ASEE Annual Conf. & Exposition*, 2019.

[4] A. Beaufort. *College Writing and Beyond: A New Framework for University Writing Instruction*. University Press of Colorado, 2008.

[5] K. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, et al. Manifesto for agile software development. 2001.

[6] M. Biagioli. From book censorship to academic peer review. *Emergences: J. for Study of Media & Composite Cultures*, 12(1):11–45, 2002.

[7] S. Blair-Goldensohn, K. Hannan, R. McDonald, T. Neylon, G. Reis, and J. Reynar. Building a sentiment summarizer for local service reviews. Technical report, Google, 2008.

[8] W. S. Cleveland. *Visualizing data.* Hobart Press, 1993.

[9] G. Domik. Who is on my team: Building strong teams in interdisciplinary visualization courses. In *ACM Siggraph ASIA Educators Program*, page 9, 2009.

[10] G. Domik. A data visualization course at the University of Paderborn. In *Pedagogy Data Visualization, IEEE VIS Workshop*, 2016.

[11] N. Elmqvist and D. S. Ebert. Leveraging multidisciplinarity in a visual analytics graduate course. *IEEE Comp. Graph. and Appl.*, 32(3):84–87, 2012.

[12] M. J. Eppler. A comparison between concept maps, mind maps, conceptual diagrams, and visual metaphors as complementary tools for knowledge construction and sharing. *Information Visualization*, 5(3):202–210, 2006.

[13] A. Friedman. Mapping subjectivity, dissensus, and concordance: Visualizing peer review and writing assessment. In *Science and Information Conf.*, 2015.

[14] A. Friedman and P. Rosen. Leveraging peer review in visualization education: A proposal for a new model. In *Pedagogy Data Visualization, IEEE VIS Workshop*, 2017.

[15] V. Garousi. Applying peer reviews in software engineering education: An experiment and lessons learned. *IEEE Trans. on Education*, 53(2):182–193, 2010.

[16] S. Gielen, E. Peeters, F. Dochy, P. Onghena, and K. Struyven. Improving the effectiveness of peer feedback for learning. *Learning and Instruction*, 20(4):304–315, 2010.

[17] J. K. Gilbert. Visualization: An emergent field of practice and enquiry in science education. *Visualization: Theory and Practice in Science Education*, pages 3–24, 2008.

[18] P. Hanrahan and K.-L. Ma. Teaching visualization. *ACM Siggraph*, 39(1):4–5, 2005.

[19] S. He and E. Adar. Vizitcards: A card-based toolkit for infovis design education. *IEEE Trans. on Vis. and Comp. Graph.*, 23(1):561–570, 2017.

[20] M. A. Hearst, P. Laskowski, and L. Silva. Evaluating information visualization via the interplay of heuristic evaluation and question-based scoring. In *ACM CHI*, pages 5028–5033, 2016.

[21] N. JD and A. Caas. The origins of the concept mapping tool and the continuing evolution of the tool. *Information Visualization*, 5(3), 2006.

[22] S. Kearney. Improving engagement: the use of 'authentic self-and peer-assessment for learning' to enhance the student learning experience. *Assessment & Evaluation in Higher Education*, 38(7):875–891, 2013.

[23] A. Kerren, J. T. Stasko, and J. Dykes. Teaching information visualization. In *Information Visualization*, pages 65–91. Springer, 2008.

[24] R. Kosara. Visualization criticism-the missing link between information visualization and art. In *Intl. Conf. on Information Visualization*, pages 631–636, 2007.

[25] L. Li, X. Liu, and A. L. Steckelberg. Assessor or assessee: How student learning improves by giving and receiving peer feedback. *British J. of Educational Technology*, 41(3):525–536, 2010.

[26] J. McGourty, P. Dominick, and R. Reilly. Incorporating student peer review and feedback into the assessment process. *Frontiers in Education Conf.*, 1, 1988.

[27] S. McKenna, A. Lex, and M. Meyer. Worksheets for guiding novices through the visualization design process. *arXiv preprint arXiv:1709.05723*, 2017.

[28] S. McKenna, D. Mazur, J. Agutter, and M. Meyer. Design activity framework for visualization design. *IEEE Trans. on Vis. and Comp. Graph.*, 20(12):2191–2200, 2014.

[29] J. Michael. Where's the evidence that active learning works? *Advances in Physiology Education*, 30(4):159–167, 2006.

[30] J. Moxley. Big data, learning analytics, and social assessment. *J. of Writing Assessment*, 6(1), 2013.

[31] R. Mulder, C. Baik, R. Naylor, and J. Pearce. How does student peer review influence perceptions, engagement and academic outcomes? a case study. *Assessment & Evaluation in Higher Education*, 39, 2014.

[32] T. Munzner. A nested model for visualization design and validation. *IEEE Trans. on Vis. and Comp. Graph.*, 15(6):921–928, 2009.

[33] T. Munzner. *Visualization analysis and design*. AK Peters/CRC Press, 2014.

[34] T. L. Naps, G. Rling, V. Almstrum, W. Dann, R. Fleischer, C. Hundhausen, A. Korhonen, and Others. Exploring the role of visualization and engagement in computer science education. In *ACM SIGCSE Bulletin*, volume 35(2), 2002.

[35] National Research Council. *How People Learn: Brain, Mind, Experience, and School: Expanded Edition*. National Academies Press, 2000.

[36] R. S. Nowacek. *Agents of integration: Understanding transfer as a rhetorical act*. SIU Press, 2011.

[37] G. S. Owen, G. Domik, D. S. Ebert, J. Kohlhammer, H. Rushmeier, B. S. Santos, and D. Weiskopf. How visualization courses have changed over the past 10 years. *IEEE Comp. Graph. and Appl.*, 33(4):14–19, 2013.

[38] E. Panadero and A. Jonsson. The use of scoring rubrics for formative assessment purposes revisited: A review. *Educational Research Review*, 9:129–144, 2013.

[39] S. G. Powell. The teachers forum: Six key modeling heuristics. *Interfaces*, 25(4):114–125, 1995.

[40] A. Renfro. Meet generation Z, 2012.

[41] C. Rohrdantz, F. Mansmann, C. North, and D. A. Keim. Augmenting the educational curriculum with the visual analytics science and technology challenge: Opportunities and pitfalls. *Information Visualization*, 13(4):313–325, 2014.

[42] H. Rushmeier, J. Dykes, J. Dill, and P. Yoon. Revisiting the need for formal education in visualization. *IEEE Comp. Graph. and Appl.*, 27(6), 2007.

[43] B. S. Santos, S. Silva, and P. Dias. Heuristic evaluation in visualization: An empirical study: Position paper. In *IEEE Evaluation and Beyond-Methodological Approaches for Visualization (BELIV)*, pages 78–85, 2018.

[44] S. Schleimer, D. S. Wilkerson, and A. Aiken. Winnowing: local algorithms for document fingerprinting. In *ACM SIGMOD International Conf. on Management of Data*, pages 76–85, 2003.

[45] J. Scholtz, C. Plaisant, M. Whiting, and G. Grinstein. Evaluation of visual analytics environments: The road to the visual analytics science and technology challenge evaluation methodology. *Information Visualization*, 13(4):326–335, 2014.

[46] A. Shamim, V. Balakrishnan, and M. Tahir. Evaluation of opinion visualization techniques. *Information Visualization*, 14(4):339–358, 2015.

[47] S. A. Sorby and B. J. Baartmans. The development and assessment of a course for enhancing the 3-d spatial visualization skills of first year engineering students. *J. of Eng. Education*, 89(3):301–307, 2000.

[48] R. Spier. The history of the peer-review process. *TRENDS in Biotechnology*, 20(8):357–358, 2002.

[49] I. Van den Berg, W. Admiraal, and A. Pilot. Design principles and outcomes of peer assessment in higher education. *Studies in Higher Education*, 31(03):341–356, 2006.

[50] I. Van den Berg, W. Admiraal, and A. Pilot. Peer assessment in university teaching: evaluating seven course designs. *Assessment & Evaluation in Higher Education*, 31(1):19–36, 2006.

[51] M. Van Zundert, D. Sluijsmans, and J. Van Merriënboer. Effective peer assessment processes: Research findings and future directions. *Learning and instruction*, 20(4):270–279, 2010.

[52] F. Vigas and M. Wattenberg. Design and redesign in data visualization, 2015.

[53] P. Walker. Let's disagree (to agree): Queering the rhetoric of agreement in writing assessment. In *Composition Forum*, volume 35, 2017.

[54] E. Wall, M. Agnihotri, L. Matzen, K. Divis, M. Haass, A. Endert, and J. Stasko. A heuristic approach to value-driven evaluation of visualizations. *IEEE transactions on visualization and computer graphics*, 25(1):491–500, 2018.

[55] E. Wardle and K. Roozen. Addressing the complexity of writing development: Toward an ecological model of assessment. *Assessing Writing*, 17(2):106–119, 2012.

[56] M. Ware. Peer review: Recent experience and future directions. *New Review of Information Networking*, 16(1):23–53, 2011.

[57] D. Weaver and A. Esposto. Peer assessment as a method of improving student engagement. *Assessment & Evaluation in Higher Education*, 37(7):805–816, 2012.

[58] M. A. Whiting, C. North, A. Endert, J. Scholtz, J. Haack, C. Varley, and J. Thomas. Vast contest dataset use in education. In *IEEE Symp. on Vis. Analytics in Sci. and Tech.*, pages 115–122, 2009.

[59] H. Wickham. A layered grammar of graphics. *J. of Computational and Graphical Statistics*, 19(1):3–28, 2010.

[60] L. Wilkinson. The grammar of graphics. In *Handbook of Computational Statistics*, pages 375–414. Springer, 2012.