Applications of Artificial Intelligence in Peer Assessment

Dr. Edward F. Gehringer, North Carolina State University

Dr. Gehringer is an associate professor in the Departments of Computer Science, and Electrical & Computer Engineering. His research interests include computerized assessment systems, and the use of natural-language processing to improve the quality of reviewing. He teaches courses in the area of programming, computer architecture, object-oriented design, and ethics in computing.

Dr. Ferry Pramudianto, North Carolina State University

Dr. Ferry Pramudianto is a Senior Research Engineer at Computer Science Department in NC State University. He has more than seven years of experience in European projects, during which he has led three multinational teams, organized technology transfer workshops, and held presentations in international conferences, as well as for the European Commission. His main research area includes Peer Assessment, Learning Analytics, Service-Oriented Architecture, Model Driven Development, and the Internet of Things.

Mr. Abhinav Medhekar, North Carolina State University
Mr. Chandrasekar Rajasekar, crajase@ncsu.edu

Master of Computer Science Student at North Carolina State University.

Zhongcan Xiao, North Carolina State University

Applications of Artificial Intelligence in Peer Assessment

Abstract

Peer assessment has at least a 50-year history in academia, and online applications for peer assessment have been available for more than 20 years. Until recently, online applications simply transmitted classmates' feedback to each other. But in the past decade, facilities have been incorporated to automatically recognize good reviews. This helps authors know which suggestions to follow and helps reviewers improve their reviews. It can also aid in assigning peer grades. Several types of data can be used to determine review quality. These metrics can be combined using machine-learning and neural-network models to produce better estimates of review quality, and hence better estimates of the quality of reviewed work. This paper discusses past work in automatically assessing reviews, and summarizes our current efforts to build on that work.

Keywords: peer review, peer assessment, peer feedback, natural language processing, convolutional neural networks, Tensorflow

1. Introduction

Peer assessment has been shown to improve student learning in disciplines all across the curriculum [1]. A large number of online applications [2] have been written to facilitate students reviewing each other's work. However, there is meager value in peer assessment unless students are careful and well informed in reviewing their classmates' work [3].

This suggests that reviews should not be taken at face value, but rather should be vetted in some way. This could be done manually, by having the course staff look at each review to assess its credibility. But this would be time consuming, and defeats one of the major advantages of peer assessment—that a large class can receive just as much feedback as a small class, with no extra investment of staff time. So, we seek an automated way to assess reviews.

Automated evaluation of reviews offers three distinct benefits..

1. Determines which reviews are useful to the author. When peer assessment is used formatively, reviews are intended to help authors revise their work. The author may receive several reviews making disparate or conflicting comments about the work. Which guidance is most important to follow? Online rating sites such as Amazon and TripAdvisor have confronted the same problem by highlighting useful reviews. A peer assessment system could do so too.

2. Determines reliable peer grades. Peer assessment is sometimes used summatively, especially in MOOCs. Usually this is done by determining the grade from some kind of

average of peer ratings. Until now, peer grading has not been accurate enough to be used routinely. If a system could automatically determine which reviews were credible, it could accord them more weight in calculating peer grades, and better match the scores that are assigned by the instructor.

3. Helps reviewers to improve their reviews. Most students are not naturally very good reviewers. They need to know what to look out for as they are reading their classmates' work, and they need to know if their review measures up to the standards of the rest of the class. Ideally, we would like to give feedback to reviewers just before they submit their review. Then they would have the opportunity to go back and improve it if it didn't measure up.

There are two steps in determining review quality. The first is to gather metrics that can be used to evaluate reviews. The second is to combine the metrics in a way that is able to approximate the way an instructor would evaluate a review. It is here that artificial intelligence can help.

2. Metrics

Peer-assessment systems normally collect both textual feedback and numerical scores, where reviewers are asked to rate various characteristics of the work on some predetermined scale. Both of these feedback modes can be used to derive metrics.

2.1 Natural language processing. Textual feedback can be assessed by natural-language processing (NLP) techniques. NLP metrics fall into three general categories. *Structure and syntax* metrics include *volume* (the number of distinct words in the review or review comment, weight or relevance like TF-IDF of each word in the review), number of sentences, average sentence length, and the fraction of sentences that are interrogative or exclamatory. NLP metrics used to assess reviews for e-commerce sites [4] also include the fraction of tokens that are nouns, verbs, adjectives and adverbs, first-person subjects, etc. *Content* metrics encompass review tone (how positive or negative the review sounds), as well as feedback features like problem detection, suggestions, and localization (which pinpoints the location of a problem or other feature) [5]. *Relevance* metrics estimate whether the review is relevant to the work being reviewed. This is done by matching text from the review with text from the work or the review rubric, taking into account the presence of hypernyms (more general words for the same thing) or hyponyms (more specific words).

2.2 Reputation systems. While NLP is helpful in measuring formative feedback (feedback given to help the author improve), reputation systems can be helpful in producing summative feedback (i.e., peer grades). A reputation system [6] is a way of comparing scores given by one reviewer with scores given by others. For example, suppose that one reviewer assigns higher scores to each artifact that he rates than other raters assign to the same artifact. Then he might be considered an "easy grader," and his scores might be adjusted downward when computing peer grades. Suppose that another reviewer assigns lower scores than her peers assign to the same

document. She might be a "hard grader," and it might be appropriate to raise those scores when computing a peer grade. Reviewers who assign similar scores to those assigned by others might be considered more reliable. This provides a way to correct for the level of difficulty encountered by an author, in the same way that strength-of-schedule ratings are used in determining rankings of FBS teams in college football.

Most reputation systems [7–9] also include a measure for "spread," the degree to which a particular reviewer rates different work differently. Suppose the average rating for all students is 4 on a scale of 1 to 5. Then a reviewer who answers "4" to every rubric item on every review might well be close to the average score received by each author. But that reviewer would not be very credible, because (s)he failed to distinguish between the quality of different pieces of work. So reviewers are given more credence if they have a higher spread.

2.3 Rejoinders, or "back-reviews." Just as reviewers rate authors' work, many online systems ask authors to rate reviews. These rejoinders can be taken as a metric for review quality, with one important caveat: There is a possibility that an author will "retaliate" for a low score from a reviewer by giving that reviewer a low score for their review. For that reason, systems tend not to use a straight average of rejoinder scores as a metric for reviews. Crowdgrader, for example, excludes the top 25% of rejoinders and the bottom 25%, and averages the remaining author ratings [10] to arrive at a metric for review quality.

2.4 Grades on earlier work. Good students tend to be competent reviewers [11], so a system might accord greater credence to reviews by students who have scored highly on previous assignments. But this may raise a practical problem, in that if the other work has been graded outside the peer-review system, the grades must be uploaded to the review system, and then any changes pursuant to grade appeals must be recorded in the peer-review system as well as the gradebook. So outside of MOOCs where everything is peer-graded, this metric is not used very often.

2.5 Calibration. Calibrated Peer Review™ [12] pioneered the approach of having each student rate sample artifacts that had previously been rated by the instructor. The student's ratings were then compared with the instructor's, and the closeness of their agreement determined the student's Reviewer Competency Index (RCI). RCI is used as a weighting factor when averaging peer-review scores to compute peer grades. Several other peer-assessment systems have since adopted the calibration approach.

2.6 Quizzing. Cursory reviews are unlikely to be accurate, so perhaps peer reviewers should have to pass a quiz showing that they understand the work they are reviewing. Authors can be asked to create a quiz, and then the reviewer's score on the quiz can also be used as an indication of competency [13].

3. Applying the Metrics

Several groups of researchers have attempted to combine subsets of these metrics to provide reliable peer assessment. Studying a Coursera MOOC on human-computer interaction, Piech, Huang, et al. [14] combined calibration data with grades given and received on earlier work. They were able to produce reliable results, but even with their best statistical model, more than a quarter of their peer grades were more than 5% away from the grade the instructor would have assigned. In a study on other Coursera MOOCS, Kulkarni et al. [15] determined that 40% of peer grades were at least one letter grade away from hypothetical instructor grades. Lee [16] used reputation systems and NLP metrics in several machine-learning models to predict instructor grades, and found that decision-tree and *k*-nearest neighbor models were most effective.

Despite these efforts, the best peer-grading models are still not accurate enough to be used in regular courses (as opposed to MOOCs, where they are accepted for reasons of exigency). Our hope is that by combining several of these metrics using various machine-learning approaches, we will be able to have an online application produce peer grades that can be used without instructor intervention, except in rare cases.

We are using machine learning in two ways. The first is in creating rubrics to guide students in peer assessing each other's work. When students rate other students' work on a specific criterion (say, Organization), that criterion exhibits a specific inter-rater reliability. High inter-rater reliability is desirable; it means that different reviewers tend to pick the same score when rating the same work. A good criterion exhibits high inter-rater reliability, and a machine-learning algorithm can "learn" the characteristics of a good criterion. Then when new criteria are added to rubrics, it can predict whether these criteria will have high or low reliability. Instructors can use this information to improve their rubric criteria. So far, our experiments indicate that criteria expressed as complete sentences have higher reliability that criteria expressed as words or phrases.

Machine learning can also be used to estimate the grade that an instructor would have assigned to a peer review. After being trained on a set of instructor-graded peer reviews, a machine-learning model can predict grades when a new review comes in. Predictions can be made in one of two ways.

- regression (where the model is trained to directly predict instructor grades e.g., 95, 90, 83, etc.), or
- classification (where grades are broken down into various ranges, e.g., 0–50, 50–65, 65–80 etc., and each range is treated as a separate class).

The quality of the textual feedback could also be used for inferring peer grades. For instance, we could have the instructor grade a set of artifacts and provide textual feedback as well. Then, using TF-IDF similarity measure, we measure the similarity of the textual feedback given by the

instructor for the sample artifact with the textual feedback given by the students [17]. The similarity score then is used to extrapolate the sample scores using regression approaches such as SVMs or decision trees.

4. Analyzing the textual feedback

Natural language processing is well suited to analyzing the structure and syntax of a text review. To identify specific features of the review text, such as whether it contains problem detection, suggestions, and localization, several methods can be adopted. For instance, Xiong & Litman, [5] propose a machine-learning method that extracts features by using regular expressions to recognize common phrases that identify localization (e.g., “on page 5”, “the section about”). They are able to identify those features with 77.4% accuracy, and both precision and recall are around 77%.

Ramachandran and Gehringer [18] propose a graph-based cohesion approach to identify those features. They represent review text by using word-order graphs. First, the review text is tagged with a POS tagger. Then, the graph is built with vertices containing tokens or phrases, connected by edges that represent dependencies such as subject-verb (SUBJ), verb-object (OBJ) or noun-modifier (NMOD). The graph is then matched with graphs extracted from review samples that have been manually labeled.

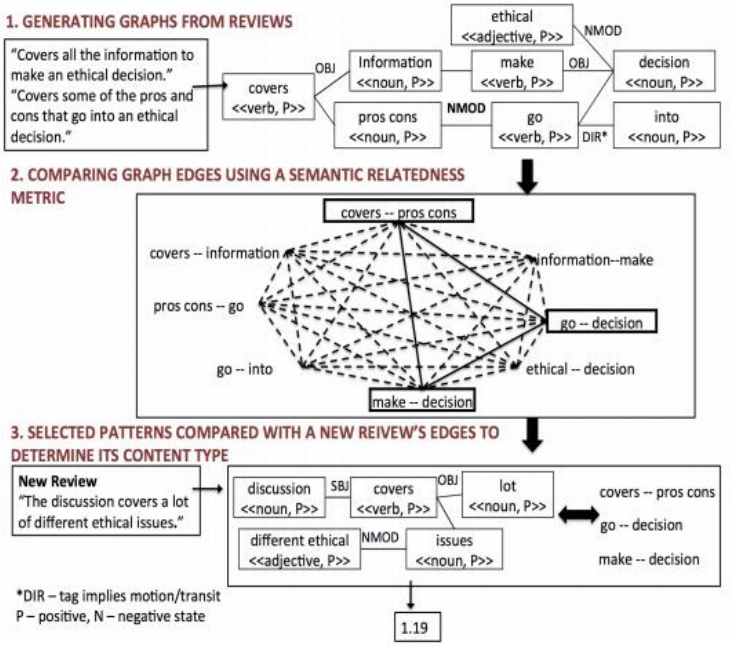

Figure 1: Illustration of graph based cohesion approach (Ramachandran & Gehringer 2015)

Another approach would be to use artificial neural networks, such as convolutional neural networks for classifying reviews. CNNs have been used successfully for computer-vision tasks, and we could try using similar approaches for our textual data as well. However, use of these

networks would require large amounts of training data. Unfortunately obtaining a large amount of training data is very expensive. We have added a feature to our Expertiza peer-assessment system [19, 20] that allows the *author* to tag whether a particular feature (e.g., helpfulness) is present in the review. For each criterion in a review, the user interface asks the author whether the comment possesses a particular feature. The interface shows the author a slider for each feature. The author can slide it to the right to indicate "yes" and to the left to indicate "no." Data from this interface will be accumulated over several semesters and used to train the model.

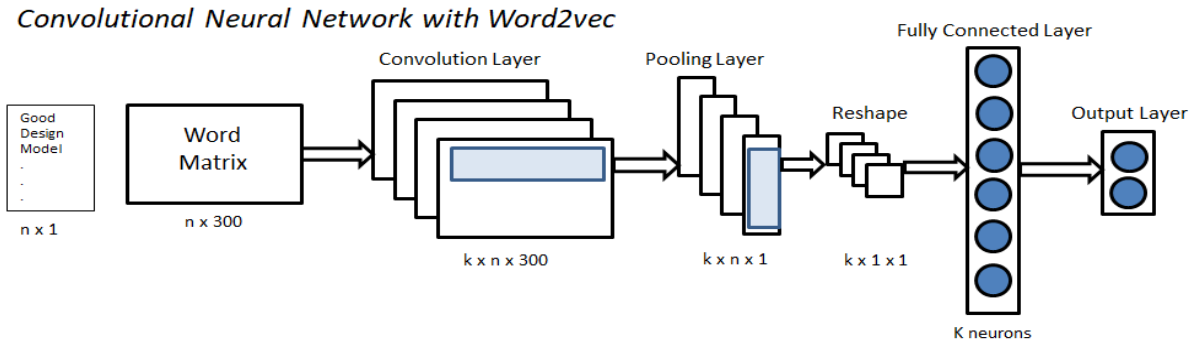

Figure 2. A Convolutional Neural Network with Word2vec

As depicted in Figure 2, neural networks can be trained using the features mentioned in Section 2. The input vector to the neural network is formed by combining the NLP metrics and word2vec or doc2vec vectors. Word2vec is used to obtain numerical vector-like representations for words so that words that are similar to each other in a particular context have similar values in the corresponding dimensions of the vector. Doc2vec is similar, but works on text strings rather than individual words. Given the tagged data, we can use deep neural networks to predict tags such as problem, praise, solution, and localization. These review tags could then be used to give feedback to the reviewer, or to approximate the grade that the instructor would have given the original review or original artifact, thus achieving any of the three benefits delineated in Section 1.

The number of layers needed in the deep neural network depends on the complexity of the data and the review tags that we want to predict. Simple tags such as identifying problems or praise could be handled using a shallow neural network, whereas complex tags like mitigation or localization requires a much deeper neural network. The amount of data required for this approach is directly proportional to the complexity and depth of the model. Thus, we keep our neural network as shallow as possible and try to improve the performance by tuning hyperparameters like dropout, learning method, and number of epochs. Finding the ideal hyperparameter is a laborious task because of the number of different combinations that need to be tried. We can search the hyperparameter space for ideal values effectively by trying random combinations of values for hyperparameter and finding the range of values at which the model performs better for each hyperparameter, and then trying all combinations within that selected range. Using neural network with our existing training set we have achieved an accuracy of

62%–74%. However, we believe that as the size of the training set increases, we will likely be able to achieve better accuracy than non-machine-learning approaches.

5. Summary

Automated evaluation of peer assessments is desirable because it can help reviewers improve their reviews, and it can assist in peer-grading either submitted work (artifacts) or reviews. Several metrics have been used for measuring the quality of reviews. Some of these are based on review text (natural language processing, for example), and others are based on peer-assigned review scores (reputation systems, for example). At least a half-dozen kinds of metrics have been tried. They are only beginning to be combined to produce more reliable evaluations. Machine learning can be applied to predict the score that an instructor would assign to a review, or to predict the usefulness of a review to an author. To provide the training set for our machine-learning algorithms, we have designed an interface to allow authors to tag review comments as containing any of a set of features. Given the importance of peer assessment as an educational practice, we should expect further progress in using automated techniques to improve the grades and feedback received by students.

6. References

1. Topping, K. J. (2009). Peer assessment. *Theory into practice*, *48*(1), 20-27.

2. Babik, D., Gehringer, E. F., Kidd, J., Pramudianto, F., & Tinapple, D. (2016). Probing the landscape: Toward a systematic taxonomy of online peer assessment systems in education. Second Workshop on Computer-Supported Peer Review in Education, associated with Educational Data Mining 2016, Raleigh, NC, June 29, 2016.

3. Ashton, S., & Davies, R. S. (2015). Using scaffolded rubrics to improve peer assessment in a MOOC writing course. *Distance Education*, *36*(3), 312-334.

4. Kim, S. M., Pantel, P., Chklovski, T., & Pennacchiotti, M. (2006, July). Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on empirical methods in natural language processing* (pp. 423-430). Association for Computational Linguistics.

5. Xiong, W., & Litman, D. (2011, June). Understanding differences in perceived peer-review helpfulness using natural language processing. In *Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications*(pp. 10-19). Association for Computational Linguistics.

6. Song, Y., Hu, Z., & Gehringer, E. F. (2015, October). Pluggable reputation systems for peer review: A web-service approach. In *Frontiers in Education Conference (FIE), 2015 IEEE* (pp. 1-5). IEEE.

7. Hamer, J., Ma, K. T. ., & Kwong, H. H. . (2005). A method of automatic grade calibration in peer assessment. In *Proceedings of the 7th Australasian conference on Computing education-Volume 42* (pp. 67–72).

8. Lauw, H. W., Lim, E. P., & Wang, K. (2007). Summarizing review scores of "unequal" reviewers. In *SIAM International Conference on Data Mining*.

9. Cho, K., & Schunn, C. D. (2007). Scaffolded writing and rewriting in the discipline: A web-based reciprocal peer review system. *Computers & Education*, *48*(3), 409–426.

10. De Alfaro, L., & Shavlovsky, M. (2014, March). CrowdGrader: A tool for crowdsourcing the evaluation of homework assignments. In *Proceedings of the 45th ACM technical symposium on Computer science education* (pp. 415-420). ACM.

11. Chiou, Y., & Shih, T. K. (2015). Auto grouping and peer grading system in massive open online course (MOOC). *International Journal of Distance Education Technologies (IJDET)*, *13*(3), 25-43.

12. Russell, A. A. (2004). Calibrated peer review-a writing and critical-thinking instructional tool. *Teaching Tips: Innovations in Undergraduate Science Instruction*, *54*.

13. Song, Y., Hu, Z., & Gehringer, E. F. (2016). Who Took Peer Review Seriously: Another Perspective on Student-Generated Quizzes. In *EDM (Workshops)* Second Workshop on Computer-Supported Peer Review in Education, associated with Educational Data Mining 2016, Raleigh, NC, June 29, 2016.

14. Piech, C., Huang, J., Chen, Z., Do, C., Ng, A., & Koller, D. (2013). Tuned models of peer assessment in MOOCs. 6th International Conference on Educational Data Mining, Memphis, TN, July 2013. *arXiv preprint arXiv:1307.2579*.

15. Kulkarni, C., Wei, K. P., Le, H., Chia, D., Papadopoulos, K., Cheng, J., & Klemmer, S. R. (2015). Peer and self-assessment in massive online classes. In *Design thinking research* (pp. 131-168). Springer, Cham.

16. Da Young Lee, F. P., & Gehringer, E. F. Prediction of Grades for Reviewing with Automated Peer-review and Reputation Metrics. Second Workshop on Computer-Supported Peer Review in Education, associated with Educational Data Mining 2016, Raleigh, NC, June 29, 2016.

17. Huang, C. H., Yin, J., & Hou, F. (2011). A text similarity measurement combining word semantic information with TF-IDF method. *Jisuanji Xuebao* (Chinese Journal of Computers), 34(5), 856-864.

18. Ramachandran, L., & Gehringer, E. F. (2015, April). Identifying Content Patterns in Peer Reviews Using Graph-based Cohesion. In FLAIRS Conference (pp. 269-275).

19. Gehringer, E., Ehresman, L., Conger, S. G., & Wagle, P. (2007). Reusable learning objects through peer review: The Expertiza approach. *Innovate: Journal of Online Education*, *3*(5), 4.

20. Gehringer, E. F. (2009). Expertiza: information management for collaborative learning. *Monitoring and Assessment in Online Collaborative Environments: Emergent Computational Technologies for E-Learning Support*, 143-159.