A Hybrid Approach for Mitigating Learners' Rogue Review Behavior in Peer Assessment

Gabriel Badea and Elvira Popescu(✉)

Computers and Information Technology Department, University of Craiova, Craiova, Romania
{gabriel.badea,elvira.popescu}@edu.ucv.ro

Abstract. Rogue reviews represent an important challenge in the application of the peer assessment process in educational settings. Students sometimes provide inappropriate evaluations of their peers' work, due to laziness, malevolence, retaliation, or dishonesty. Various approaches have been described in the literature for mitigating personal bias and increasing assessment reliability. The current paper proposes an innovative mechanism for managing rogue reviews, based on a hybrid approach: combining automatic labelling of suspicious reviews by the system with manual analysis of their content by the teacher. In addition, dedicated prompts are displayed to the students, providing specific recommendations for revising potentially rogue reviews. The mechanism was integrated in an existing peer evaluation system called LearnEval. The platform was used in a pilot study whose results are reported and discussed in the paper; several lessons learned from the experience and potential improvements are also included.

Keywords: Peer evaluation · Peer assessment platform · Rogue reviews · Feedback quality

1 Introduction

Peer assessment is an educational activity applied in a wide range of areas, both formally and informally. The feedback offered by the students is a critical component for learning, being helpful in various ways: it provides multiple perspectives for authors to consider [10], each learner receives individualized feedback [6], it encourages reflection and metacognition, and fosters critical thinking skills [14]. Some practitioners consider the timeliness and diversity of the feedback even more important than its content [10]; however, ensuring a quality peer feedback is an important endeavor [7, 10, 16].

A challenging aspect in the successful application of the peer assessment process is represented by the rogue behavior of the students. Rogue reviews are inappropriate evaluations that have as source laziness, malevolence, retaliation, or dishonesty [15], affecting students' learning enthusiasm [17]. The rogue assessors assign arbitrary grades regardless of the solution quality [15]; indeed, in every course it can be assumed that a part of the learners will assign random grades during the assessment process [9]. In this context, several approaches have been proposed in the literature to mitigate the personal bias and increase assessment reliability: use of multiple reviewers for the same solution

S. Crossley and E. Popescu (Eds.): ITS 2022, LNCS 13284, pp. 24–35, 2022.
https://doi.org/10.1007/978-3-031-09680-8_3

to increase the accuracy [15, 17]; let evaluators see how other peers have assessed the same solutions they reviewed to increase self-awareness [8]; use of design prompts that alert assessors to change their reviewing standards when they submit rogue evaluations [17]; use of advanced and innovative systems that prevent rogue review behavior from the start [15].

In this paper we propose a novel hybrid approach, where automatic machine labelling is combined with manual human marking to facilitate the detection of rogue reviews, by considering aspects related to time allotted, grades assigned, or feedback content. We report on the integration of the rogue review management mechanism in an existing peer evaluation system called LearnEval, offering details regarding the implementation, context of use, a pilot study, and an initial analysis of the potentially rogue reviews. LearnEval [1, 2] is an innovative peer assessment platform that has been applied in several scenarios, especially in Project-Based Learning (PBL) settings [3]. The system allows students to assess their peers' artifacts by assigning grades and providing feedback; the students are guided during the assessment using criteria defined by the instructor; advanced mechanisms for reviewer calibration, monitoring and visualizations as well as dynamic review allocation are included in the platform.

The paper is structured as follows. Section 2 presents related work about peer assessment reliability and rogue reviewing. Section 3 describes the proposed hybrid mechanism for detecting and preventing rogue review behavior. Section 4 reports the results obtained in a pilot study performed with 52 undergraduate students. Section 5 contains a discussion about the findings, draws some conclusions and offers perspectives for future work.

2 Related Work

In the following, we present relevant works that address techniques for detecting rogue review behavior and examine peer assessment reliability.

Various ways for increasing feedback quality and avoiding the clustering of grades are presented in an early paper [7]. Several approaches are proposed, such as: deny credit unless submitting the evaluations, prevent access to one's own received feedback until the student provides feedback to peers, compute student's grade based on the grades received by the peers they reviewed (thus encouraging highly useful evaluations), and include an additional phase where the learner assesses the reviews performed by others. In addition, the accuracy of grading could be increased by requesting learners to complete a pre-certification test before evaluating their peers. Furthermore, the authors propose additional methods for preventing the clustering of grades, such as: use ranking instead of grading, or assign each student a limited number of shares that have to be distributed among the reviewed solutions.

A study to assess the accuracy and effectiveness of distributed peer assessment and to determine how often issues such as rogue reviews arise is carried out in [15]. Several factors that foster the rogue behavior are addressed, such as: laziness, retaliation, collusion, and competition. The authors use incentives to mitigate such behavior. For instance, students are demanded to provide the assessments prior to seeing their solution score. On the other hand, laziness is counteracted by reckoning the assessment process as a course activity and tailoring the student workload appropriately. An analysis was conducted to assess the level of retaliation by correlating the review scores with the ones assigned by the authors to evaluations and the value found was low. Several cases of collusion were identified, where pairs of learners settled to assign each other high scores. However, very few proofs of rogue behavior were detected in practice. Overall, the results highlight that peer assessments are accurate in comparison to a recognized standard of evaluation.

An interesting approach which relies on analyzing the lexical sophistication of feedback comments in a large-scale study is reported in [10]. The paper examines reviewers' competence in making appropriate evaluations and the complexity of the feedback provided. The textual complexity is assessed by employing five metrics: comment length, number of distinct tokens, median word length, word frequency, and average token-type ratio. The findings show that high performing students generally write more and have a better vocabulary than the rest. In addition, the analysis of lexical sophistication highlights the fact that reviewers generally produce less complex comments than instructors. However, the authors suggest that the gap could be reduced by assigning multiple evaluators for a single submission. Thus, the effect of rogue reviews is slightly lessened by assigning grades to solutions computed based on a weighted average of the received marks.

A different mechanism is proposed in PeerStudio platform [12], which shows reviewers short tips based on the feedback they provide. These helpful tips are generated using a list of relevant words extracted from the draft submitted by the student and the assignment description. Furthermore, the system uses the number of relevant words in the feedback and its length to propose enhancements. A large amount of low-quality comments is detected by employing this simple heuristic. The platform guides the reviewer to provide the most relevant feedback for the current state of the submission by internally computing the solution quality (low, medium, high). Additionally, the authors can evaluate the received assessments and deliver messages to the staff. The approach was quite successful, with students considering 45% of the comments to be "somewhat concrete or better", offering links to helpful resources, or suggestions on how to improve the solution (while the rest of the comments were simply praise or support messages).

Statistical measures are also proposed in [17] for detecting non-consensus and radical review behavior in peer assessment. Non-consensus occurs when multiple reviewers disagree and have contrary opinions on the same aspect. Moreover, a part of the reviewers have radical behavior during assessment by repeatedly assigning low or high grades, without considering the actual solution quality. In this context, the EduPCR peer assessment platform automatically identifies non-consensus by means of the standard deviation of the grades assigned by reviewers. Teacher arbitration is triggered when such non-consensus is discovered in a group of reviewers. On the other hand, a reviewer is marked by the platform as a radical candidate when they repeatedly offer high or low grades. In such cases, a short message or an email is delivered to the instructor, who then manually inspects the grades assigned by that reviewer.

Furthermore, some papers propose innovative ways to increase peer assessment reliability. For instance, fuzzy logic is used in [4]; the approach is employed to model opinions, the opinions are further weighted based on their validity, and in the end, they are aggregated to achieve a reliable process. Automatic classifiers are used in [13] to evaluate the quality of the peer assessment process, and various metrics for gauging the reliability and validity of the reviewers are presented. Paper [5] presents the hypothesis that allowing both reviewers and solution authors to introduce themselves and exchange messages with each other during the peer assessment process rises the feedback quality. Finally, game-like elements are employed in [11] to incentivize students to provide reliable assessments.

The current paper adds to the literature by proposing a novel mechanism for managing rogue reviews, which is based on a hybrid approach: combining automatic labelling of suspicious reviews by the system with manual analysis of their content by the teacher. The automatic detection is based on a quantitative approach, computing a score based on a set of factors which could indicate a potentially rogue review. An additional mechanism is proposed for encouraging students to carefully check their reviews and make appropriate revisions before submitting them, as described in the next section.

3 A Mechanism for Mitigating Rogue Review Behavior

The starting point of our approach is an existing peer assessment system, called LearnEval, which we proposed in [1, 2]. The platform supports a highly configurable assessment scenario, providing several functionalities for both students and instructors: calibration module, open learner model, comprehensive monitoring and visualization features, dynamic review allocation module. In this section we report on the design and integration of a mechanism for detecting and preventing rogue review behavior. More specifically, a hybrid approach is proposed, which combines automatic appraisal of the reviews by the system with human judgment. The system tags specific reviews as potentially rogue, considerably reducing the amount of reviews that need to be manually checked by the teacher. In addition to this detection mechanism, a prevention mechanism is also put in place: warning and recommendation messages are displayed to the students when they try to submit potentially rogue reviews, which can be used to improve the review content.

A quantitative approach is used for assessing the likelihood for a review to be rogue. More specifically, a *Rogue Score* is computed every time a review is submitted, considering various criteria related to: time required for performing the evaluation, grades assigned to the assessment criteria, and quality of the feedback provided. Two scores are stored for each review: an *Initial Rogue Score* computed when the student aims to send the evaluation for the first time, and a *Final Rogue Score* computed when the evaluation is actually submitted (after the student has the chance to revise/improve it, taking into account the recommendations provided by the system).

Based on our own experience as well as a review of the literature, we extracted a set of 14 criteria that could indicate a rogue evaluation. These criteria are described in Table 1, including an impact level for each of them (i.e., criterion score). The *Rogue Score* of a review is computed by adding the corresponding scores for each fulfilled criterion. Starting from this score, a set of features are provided by the rogue review mitigation mechanism in LearnEval, both for the student and the teacher, as described in the following subsections.

Table 1. Set of criteria indicating a potentially rogue review, integrated in LearnEval

Category	Criterion description	Criterion Score
Time	Review submitted in less than five minutes after it was assigned	100
Grade	Student gives the same grade for all the reviews performed for the current assignment	100
	Student gives the same maximum grade (10) for all the reviews performed for the current assignment	100
	Student gives the same minimum grade (1) for all the reviews performed for the current assignment	50
	Student gives the same median grade (7) for all the reviews performed for the current assignment	50
Feedback	Student provides similar feedback (over 90% textual similarity) for two different assessment criteria of the current review	50
	Student provides similar feedback (over 90% textual similarity) for at least four different assessment criteria (belonging to different reviews of the same assignment)	100
	Student provides similar feedback (over 90% textual similarity) with the one written by a peer, for at least three different assessment criteria	50
	Student provides feedback that contains only whitespaces	50
	Student provides feedback that does not contain any letters or digits	50
	Student provides feedback that contains less than five words	75
	Student provides feedback that contains less than five distinct words	50
	Student provides feedback that contains repeating consecutive words (at least 4 times the same word)	25
	Student provides feedback that contains repeating consecutive letters (at least 5 times the same letter)	15

3.1 Student Perspective

The rogue review prevention mechanism automatically verifies each evaluation when the student aims to submit it (i.e., clicks the submit button). In case at least one of the rogue criteria from Table 1 is satisfied, the platform prevents the submission and displays a notification message to the student, asking them to recheck the review. A list of revision recommendations, corresponding to each fulfilled criterion, is provided to the learner, as illustrated in Fig. 1. The student has the opportunity to perform the suggested improvements and resubmit the review, after explicitly acknowledging that the review was appropriately checked and revised.

However, if the student chooses to submit a review which still fulfills the rogue criteria and is marked as such by the teacher, then their reviewing skills score is automatically

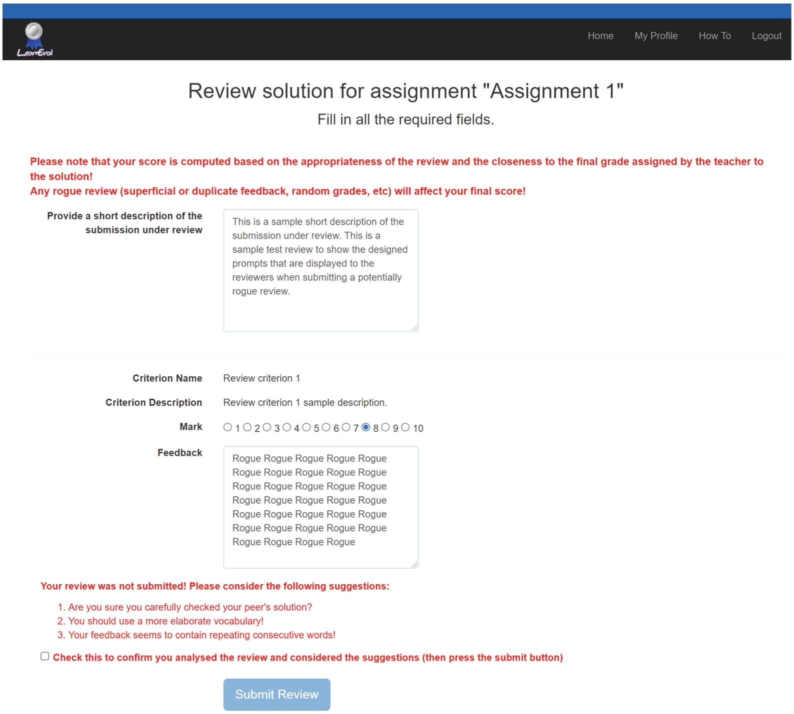

Fig. 1. LearnEval rogue review prevention module: revision recommendations displayed to the student when submitting a potentially rogue review

decreased. More specifically, LearnEval models the learner's assessment capabilities by computing an aggregated score:

$$\begin{aligned} &\mathit{ReviewingScore} = \mathit{PeerBackReviewsAvg} * \mathit{wm}1 + \mathit{TeacherBackReviewsAvg} * \mathit{wm}2 \\ &+ \mathit{AgreementWithFinalMark} * \mathit{wm}3 + \mathit{CalibrationScore} * \mathit{wm}4 - 0.5 * \mathit{RogueReviewsCount} \end{aligned}$$

where: *PeerBackReviewsAvg* represents the mean grade of the back-reviews received by the student from the solution authors, *TeacherBackReviewsAvg* represents the mean grade of the back-reviews received by the student from the teacher, *AgreementWithFinalMark* depicts the accuracy of the grades assigned by the student, *CalibrationScore* depicts student's reviewing capabilities at the start of the peer assessment process and *RogueReviewsCount* represents the number of reviews marked as rogue by the teacher (each entailing a penalty of 0.5 points); the weights are configured by the teacher, such that $wm1 + wm2 + wm3 + wm4 = 1$.

3.2 Teacher Perspective

The platform provides a dedicated *Reviews* page which allows the teacher to readily visualize all the submitted evaluations and their rogue scores (as illustrated in Fig. 2).

The instructor can order the reviews based on their *Rogue Score*, thus easily identifying the evaluations which are more likely to be rogue and focus their efforts on checking those first. In addition, detailed information is available about each student evaluation (as illustrated in Fig. 3), such as: time in review (interval between review assignment and review submission); grade assigned in back-review by the solution author; difference between the grade assigned by the reviewer and the final solution grade; initial and final *Rogue Score* along with a description of the fulfilled criteria (if any). The teacher can use this information, together with the actual content of the evaluation, to decide whether a review is indeed rogue. Once marked as rogue by the instructor, that review is no longer taken into account when computing the final grade of the solution.

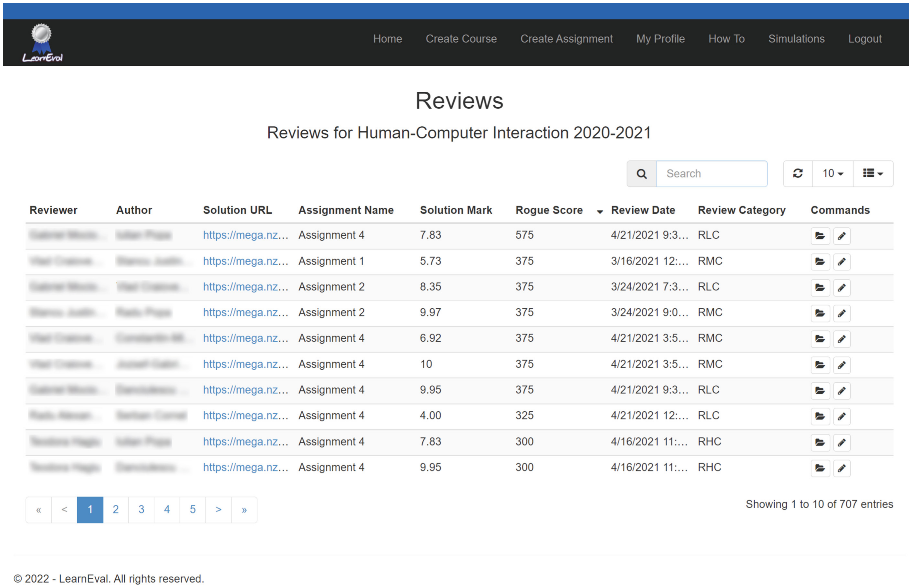

Fig. 2. LearnEval rogue review prevention module: teacher view of the reviews and their rogue score

4 Rogue Reviews Analysis: Pilot Study

4.1 Context of Study

LearnEval peer assessment platform integrating the proposed rogue review prevention mechanism was applied in the context of a *Human-Computer Interaction* course at the University of Craiova, Romania. The course followed a project-based learning approach, being taught to 52 4th year students, during the second semester of 2020–2021 academic year.

The task of the project was to design, build, and assess the user interface of a web application. The project was split in four milestones: the first deliverable referred to user modeling and storyboarding tasks; the second deliverable required students to build low

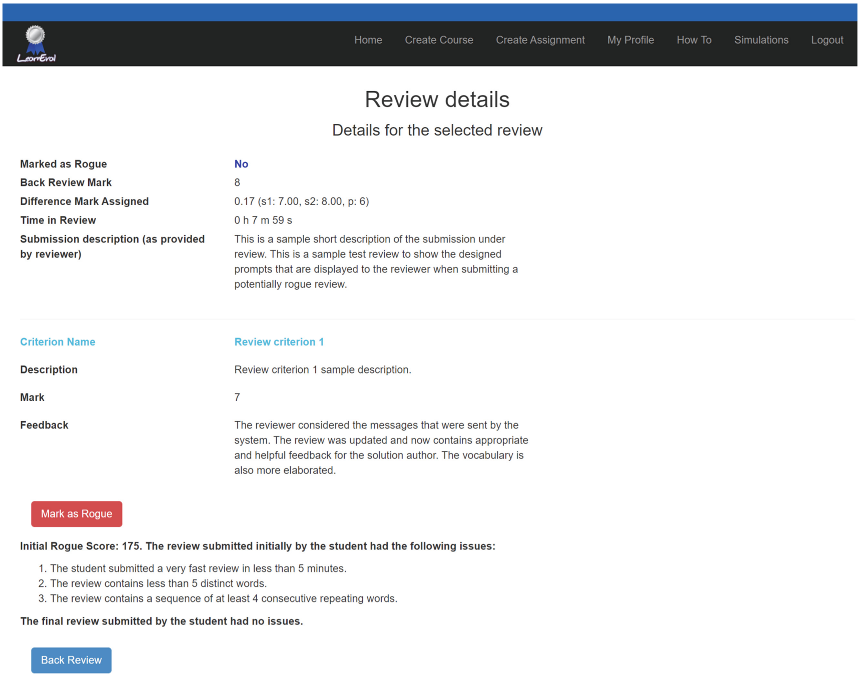

Fig. 3. LearnEval rogue review prevention module: teacher detailed view of a potentially rogue review

and high fidelity prototypes; the third deliverable required learners to implement the actual user interface; and the last one involved interface evaluation and usability testing.

The milestones of the project were used to create four peer assessment sessions. A session had a submission phase, where students submitted solutions (deliverables), followed by a review period. The review period was further split in two stages: a first review phase, where students had to mandatory assess three peers' solutions, and an extra review phase, where students were able to optionally assess up to three peers' solutions. The learners were granted bonus points at the end of the course based on the number of extra solutions reviewed. The assessment criteria were defined by the teacher and varied according to the requirements of the milestone. The student had to assign a grade on a scale from 1 to 10 and provide feedback for each criterion. Furthermore, a short summary of the solution was required. At the beginning of the semester students were provided with an introductory meeting in which the instructor explained the peer assessment process, including a description of the rogue review behavior, its consequences and why it should be avoided.

4.2 Results Analysis

In the following, we analyze the *Rogue Scores* values computed by the system for potentially rogue evaluations. A total of 707 reviews were submitted by the students.

Almost a quarter of these peer evaluations (i.e., 172 reviews), met at least one rogue review criterion from Table 1, thus requiring teacher's attention.

Most of these reviews resided in the [100, 200) score range, as illustrated in Table 2. The most common causes encountered were: many evaluations were sent very quickly by the students, in less than 5 min (criterion counting 100 points); and many reviewers provided similar feedback for at least two different assessment topics (criterion counting 50 points).

Table 2. Distribution of reviews based on *Rogue Score* (the percentages are computed out of the total number of reviews identified as potentially rogue by the system – i.e., 172 reviews)

Rogue score	[1, 100)	[100, 200)	[200, 300)	[300, 400)	400+
Reviews count	18 (10%)	107 (62%)	36 (21%)	10 (6%)	1 (<1%)

The teacher manually checked each potentially rogue review and marked the truly rogue ones correspondingly. Overall, 23% of the potentially rogue reviews were found to be actually rogue by the instructor, as detailed in Table 3. As can be seen, the percentage of evaluations identified as rogue by the teacher increases as the *Rogue Score* increases, thus, a higher score raises the likelihood for an assessment to be actually rogue. A significant raise in the percentage can be noticed starting with [200, 300) interval; the rogue likelihood of an evaluation with a score of at least 200 is more than 50% - hence this could be considered a cut-off point, above which additional measures could be taken by the system.

Table 3. Distribution of teacher-identified rogue reviews based on *Rogue Score* (the percentages are computed out of the number of reviews identified as potentially rogue by the system for each interval, as shown in Table 2)

Rogue score	[1, 100)	[100, 200)	[200, 300)	[300, 400)	400+	Total
Reviews count	1 (6%)	11 (10%)	20 (56%)	6 (60%)	1 (100%)	39 (23%)

In addition to the inadequate feedback, the grades provided in the rogue reviews were also less accurate. The average difference between the grade assigned by the student and the final solution grade in the reviews identified as potentially rogue by the system (but not by the teacher) was 0.81; as expected, this value is lower than in case of reviews identified as rogue by the teacher (i.e., 1.11). Furthermore, the correlation between the grade assigned by the student and the grade assigned by the instructor in the reviews identified as potentially rogue by the system (but not by the teacher) was 0.78, which is higher than the correlation in case of reviews identified as rogue by the teacher (i.e., 0.49).

While the rogue review detection process was successful, the review improvement component did not work as expected. Although the students were provided with review revision suggestions, as described in Sect. 3.1, the learners rarely took advantage of these recommendations before resubmitting their reviews. More specifically, only 6 reviews were revised according to the system suggestions and only 2 of them were substantially improved. Therefore, additional measures and better incentives must be devised to motivate reviewers to enhance their feedback quality.

5 Discussion and Conclusion

The paper presented the integration of a rogue review mitigation mechanism in LearnEval peer assessment platform. The mechanism applies a hybrid approach where automatic machine labelling of potentially rogue reviews is combined with teacher's manual check. The advantages of the system are twofold: on one hand, the platform significantly decreases the time spent by the teacher to identify rogue reviews, by already flagging suspicious evaluations and thus reducing the search space. On the other hand, improvement recommendations are automatically displayed to the students, who have the opportunity to revise their reviews before submitting them, which could lead to a lower number of rogue reviews.

The mechanism was employed in a pilot study involving 52 students, in the context of a Human-Computer Interaction course. The system flagged around a quarter of the evaluations as potentially rogue, which significantly decreased teacher's workload. Furthermore, an initial analysis showed that the higher the *Rogue Score*, the higher the likelihood that the review was actually rogue and marked as such by the instructor; this comes as a validation of our detection mechanism and the proposed rogue criteria. Nevertheless, the scores for some criteria could be revised in light of our initial findings; an even better approach would be to make these scores configurable by the teacher, based on the specificities of each instructional scenario. In addition, a more advanced mechanism could be envisioned, based on an extended list of criteria and a more complex fuzzy logic approach.

A limitation of our study was that very few students actually followed the recommendations provided by the system in order to improve their reviews. Various approaches could be applied to address this issue. First of all, the current mechanism only displays revision suggestions, but does not prevent the submission of the review if these suggestions are not followed. Therefore, a more complex, layered approach could be proposed, based on the value of the *Rogue Score*, such as:

1. Since the percentage of reviews identified as rogue by the teacher was low in the interval [0, 200), the flow could be kept unchanged below this threshold (i.e., simply prompt the student to consider the recommendations made by the system).
2. In the next interval, [200, 300) range, more than half of the evaluations were identified as rogue by the teacher, thus more restrictive measures could be applied. Hence, the submission of the review should not be allowed in case its final *Rogue Score* is the same as the initial one (i.e., the student needs to address at least one of the rogue criteria). Furthermore, an automatic notification could draw teacher's attention to immediately check the potentially rogue review.

3. In the following score intervals (i.e., >=300), a majority of the evaluations were marked as rogue by the teacher. Therefore, the student should not be allowed to submit a review with a *Rogue Score* over 300. However, an appeal mechanism should be made available to the student, who could ask for teacher's evaluation in case they consider their review to be a valid one.

It should be noted that the above thresholds are inferred from the current pilot study and may not be generally valid. Therefore, a configurable approach could be envisioned, in which the teacher can set the thresholds based on the particularities of the peer assessment scenario and the reviewing skills of the students. Furthermore, dedicated incentives for fostering students' motivation and encouraging them to provide higher quality feedback need to be integrated in the system. Finally, we aim to apply the improved version of the LearnEval platform in more courses and instructional scenarios and conduct a more in-depth analysis of the peer assessment quality.

References

1. Badea, G., Popescu, E.: A web-based platform for peer assessment in technology enhanced learning: student module prototype. In: Proceedings of ICALT 2019, Maceió, Brazil, pp. 372–374. IEEE Computer Society Press (2019). https://doi.org/10.1109/ICALT.2019.00115
2. Badea, G., Popescu, E.: Instructor support module in a web-based peer assessment platform. In: Proceedings of ICSTCC 2019, Sinaia, Romania, pp. 691–696. IEEE (2019). https://doi.org/10.1109/ICSTCC.2019.8885687
3. Badea, G., Popescu, E.: Using peer assessment in conjunction with project-based learning: a comparative study. In: Proceedings of ICALT 2020, pp. 376–380. IEEE Computer Society Press (2020). https://doi.org/10.1109/ICALT49669.2020.00119
4. El Alaoui, M., El Yassini, K., Ben-Azza, H.: Peer assessment improvement using fuzzy logic. In: Ben Ahmed, M., Boudhir, A.A., Younes, A. (eds.) SCA 2018. LNITI, pp. 408–418. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-11196-0_35
5. Gamage, D., Whiting, M., Perera, I., Fernando, S.: Improving feedback and discussion in MOOC peer assessment using introduced peers. In: 2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), pp. 357–364 (2018). https://doi.org/10.1109/TALE.2018.8615307
6. Gehringer, E.F.: Strategies and mechanisms for electronic peer review. In: 30th Annual Frontiers in Education Conference (FIE 2000), pp. F1B/2–F1B/7 (2000). https://doi.org/10.1109/FIE.2000.897675
7. Gehringer, E.F.: Assignment and quality control of peer reviewers. In: ASEE Annual Conference and Exposition, American Society for Engineering Education, Albuquerque, USA (2001)
8. Gehringer, E., Ma, K., Duong, V.: What peer-review systems can learn from online rating sites. In: State-of-the-Art and Future Directions of Smart Learning, pp. 341–350 (2016). https://doi.org/10.1007/978-981-287-868-7_42
9. Hamer, J., Ma, K.T.K., Kwong, H.H.F.: A method of automatic grade calibration in peer assessment. In: Proceedings of the Seventh Australasian Computing Education Conference (ACE2005), Vol. 42 of Conferences in Research and Practice in Information Technology, Newcastle, NSW, Australia, Australian Computer Society, pp. 67–72 (2005)

10. Hamer, J., Purchase, H.C., Denny, P., Luxton-Reilly, A.: Quality of peer assessment in CS1. In: Proceedings of the Fifth International Workshop on Computing Education Research Workshop, Berkeley, CA, USA, 10–11 August 2009, pp. 27–36 (2009). https://doi.org/10.1145/1584322.1584327
11. Indriasari, T.D., Luxton-Reilly, A., Denny, P.: Investigating accuracy and perceived value of feedback in peer code review using gamification. In: Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education, pp. 199–205 (2021). https://doi.org/10.1145/3430665.3456338
12. Kulkarni, C.E., Bernstein, M.S., Klemmer, S.R.: PeerStudio: rapid peer feedback emphasizes revision and improves performance. In: Proceedings of the Second ACM Conference on Learning @ Scale, Vancouver, BC, Canada, pp. 75–84 (2015). https://doi.org/10.1145/2724660.2724670
13. Molina-Carmona, R., Sattore-Cuerda, R., Compañ-Rosique, P., Llorens-Largo, F.: Metrics for estimating validity, reliability and bias in peer assessment. Int. J. Eng. Educ. **34**(3), 968–980 (2018)
14. Politz, J.G., Patterson, D., Krishnamurthi, S., Fisler, K.: CaptainTeach: multi-stage, in-flow peer review for programming assignments. In: Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education (ITiCSE 2014), pp. 267–272. ACM, New York (2014). https://doi.org/10.1145/2591708.2591738
15. Reily, K., Finnerty, P.L., Terveen, L.: Two peers are better than one: Aggregating peer reviews for computing assignments is surprisingly accurate. In: Proceedings of the ACM 2009 International Conference on Supporting Group Work, ACM, Sanibel Island, Florida, USA, pp. 115–124 (2009). https://doi.org/10.1145/1531674.1531692
16. Staubitz, T., Petrick, D., Bauer, M., Renz, J., Meinel, C.: Improving the peer assessment experience on MOOC platforms. In: Proceedings of the Third ACM Conference on Learning @ Scale, ACM, Edinburgh, Scotland, UK, pp. 389–398 (2016). https://doi.org/10.1145/2876034.2876043
17. Wang, Y., Liang, Y., Liu, L., Liu, Y.: A multi-peer assessment platform for programming language learning: considering group non-consensus and personal radicalness. Interact. Learn. Environ. **24**(8), 2011–2031 (2016). https://doi.org/10.1080/10494820.2015.1073748