IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 21, 2020, accepted December 2, 2020, date of publication December 8, 2020,
date of current version December 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3043291

Human-Machine Hybrid Peer Grading in SPOCs

YONG HAN, WENJUN WU, YITAO YAN, AND LIJUN ZHANG
State Key Laboratory of Software Development Environment, School of Computer Science, Beihang University, Beijing 100191, China

Corresponding author: Wenjun Wu (wwj@nlsde.buaa.edu.cn)

This work was supported in part by the NSFC under Grant 61532004, in part by the National Key Research and Development Program of China under Grant 2018YFB1004502, and in part by the State Key Laboratory of Software Development Environment under Grant SKLSDE-2017ZX-04.

ABSTRACT Peer grading, allowing students/peers to evaluate others' assignments, offers a promising solution for scaling evaluation and learning to massive open online courses (MOOCs) and small private online courses (SPOCs). In the environment of MOOCs, due to the varied skill levels and attitudes of online students, it is not easy for the students to present fair and accurate scores for their peers' assignments. Recently, statistical models have been proposed to improve the fairness and accuracy of peer grading, and these models have achieved good performance in MOOCs. However, our experiments demonstrate that these models fail to deliver accurate inferences in the SPOC scenario because affinity among students may seriously affect the objectivity and reliability of students in the peer-assessment process. To address this problem in SPOCs, this paper proposes a human-machine hybrid peer-grading framework, in which an CNN(Convolutional Neural Networks)-based automated grader works as a filter to ensure reasonable peer scores before Bayesian models are utilized to infer the true scores. This framework can significantly eliminate the severely biased scores of undutiful students and, thus, improve the accuracy of the true-score estimation of the Bayesian peer-grading models. Both the simulated and actual peer-grading datasets in our experiments demonstrate the effectiveness of this new framework for SPOCs.

INDEX TERMS Peer grading, human-machine hybrid framework, Bayesian model, automated grader.

I. INTRODUCTION

Small private online courses (SPOCs) are a version of massive open online courses (MOOCs) that are used locally by on-campus students. SPOCs often have a relatively smaller number of students than MOOCs, and SPOC students may come from the same classroom and know each other, while MOOC participants are often distributed all over the world. Despite the differences between SPOCs and MOOCs, a SPOC course follows the same peer-grading (PG) process as a MOOC course when an instructor must evaluate hundreds of open-ended essays and exercises such as mathematical proofs and engineering design problems within a deadline. Given the scale of the submissions in such a course, if the course instructor and his or her teaching assistants (TAs) had to examine all the submissions by themselves, these instructors would be overwhelmed by the grading workload and not have sufficient time or energy for other teaching tasks that are helpful for students.

The associate editor coordinating the review of this manuscript and approving it for publication was Nadeem Iqbal.

Previous research efforts on PG suggest that there is significant disparity between the scores presented by student graders and the true scores given by an instructor. This disparity occurs because students sometimes cannot perform grading tasks in a manner of a professional instructor with the right skills and dedication. Therefore, how to correctly aggregate peer-assessment results to generate a fair score for every homework submission is a major challenge.

In the PG process of SPOCs, every student grader needs to submit his or her answer to the problems of home assignments and evaluate their peers' submissions according to the rubrics provided by the course instructor. The PG system collects the peer scores of each student's submission and estimates the true score for the submission using advanced statistical models [1], [2]. These models mainly consider the factors affecting the aggregation of peer scores including the bias and reliability of every student grader. These factors and the true scores of the student submissions are described as latent random variables to build a Bayesian model to infer the final scores of student submissions.

However, these PG algorithms, mostly designed for MOOCs courses, may perform poorly in SPOC courses.

Because of the affinity among SPOC students, these students tend to assign random scores to other submissions without seriously evaluating their peers' homework. Even worse, in our practical experiment, we found that certain students simply gave a full score to every submission assigned to them. Therefore, such an undutiful grading behavior violates the basic assumption in those Bayesian statistical models and unavoidably generates data noises that severely reduces the performance of the models. Our simulation and real experiments confirm that the models produce inaccurate estimations for final scores in the process of PG [3].

The contribution of this paper is mainly to analyze the shortcomings of previous statistical models in the peer grading and propose an effective method. To address this problem, this paper proposes a novel human-machine hybrid framework that combines the assessment efforts of both humans and machines for PG. The framework uses a document classifier as an automated grader that evaluates students' submissions to estimate the scores and compares the calculated scores with the PG scores. We explore two methods to design the automated graders, which are based on probability-based naive Bayes and neural network-based CNN. The framework attempts to filter out the unreasonable PG scores that are significantly different from their estimations and retain the legitimate scores for the statistical models. In this way, this model can alleviate the negative impact of random student grading behavior and improve the overall performance of PG models. To verify our proposed framework, we conduct extensive experiments on actual and simulated datasets. The experimental results demonstrate that our hybrid framework outperforms the original PG models in terms of the true-score estimation accuracy without placing too much extra workload on course TAs.

The rest of the paper is organized as follows: Section 2 discusses work related to our research. Section 3 elaborates the main problems in the PG of SPOCs and explains the motivation of combining the machine and human efforts in PG. Section 4 describes the rubrics of the course in our peer grading experiment. Section 5 describes the design of the human-machine hybrid framework for PG in detail. Section 6 presents our experimental results.

II. RELATED WORK

The focus of this paper is to combine the power of human graders and a machine grader to improve the predictive ability of existing PG models. Numerous papers have been published in the field of PG research [1], [2], [5]–[8]. Most researchers attempt to tackle PG problems from two aspects: statistical methods for accurately inferring true scores and incentive mechanisms to motivate and regulate student grading behaviors.

One of the major research ideas in PG is to build a Bayesian statistical model that can accurately infer the true scores of student submissions. Such models were proposed in [1] and [2] for PG in MOOC courses with the bias and reliability of student graders as the major latent factors. In [1], three PG models named by PG1, PG2 and PG3 are defined to exploit the linear dependency between a student grader's reliability and the true score. In [2], the authors extend PG1-PG3 to define two new modes named PG4 and PG5 by assuming that a grader's reliability should follow a Gaussian or Gamma distribution with the true score as a parameter. In [4], Ueno adopts item response theory (IRT) to model the score estimation, difficulty of the problem and a grader's capability as parameters in the IRT equation. The major limitation of these models lies upon their common assumption that every student follows a statistical model in the PG process. However, in practice, especially in SPOCs, the grading behavior is heavily affected by their motivation and attitude toward PG tasks [12]. According to our research, many students often perform grading tasks in an undutiful manner and give random scores to assignments. Such grading behavior clearly violates the statistical assumptions and leads to high prediction errors for the Bayesian-based PG models.

The problem of student grading behavior has received attention from academic researchers in the field of game theory [5]–[8], [12]. Recently, the peer-prediction mechanism has been proposed to incentivize truthful reports from individual students in the process of PG. By checking the stochastic correlation between students' evaluation reports, peer prediction can reward each student if his/her report is statistically consistent with those of his/her peers. Ideally, such a mechanism designed in a PG system can effectively provide motivation to students to carefully formulate evaluations and to report evaluations honestly [5]. However, peer prediction has inherent limitations because there are potentially multiple equilibria at which students might be able to coordinate to avoid penalty without truthfully revealing an informative signal. Even when the peer prediction mechanisms do offer a truthful equilibrium, these mechanisms also always induce other uninformative equilibria [13]–[15]. In the settings of SPOCs, affinity among students makes collusion to cheat the peer prediction mechanisms highly possible in the PG process [6]. To remedy this problem, researchers introduced spot-checking mechanisms [7], [8], [17], in which the instructor and class TAs randomly check certain assignment grading reports and offer a reward to students who finish their grading task diligently. Spot-checking mechanisms enhance the performance of peer prediction by incurring greater workloads for the instructor and TAs. However, it is still uncertain how a TA should choose the reports for spot-checking.

The Bayesian-based probabilistic model relies heavily on priori values when aggregating peer scores and cannot accurately aggregate scores if the grader's factors such as the grader reliability and bias estimates are inaccurate. Therefore, the simple use of probability models in PG is not sufficient for predicting accurate peer scores. We propose a framework based on a combination of people and machines to improve the accuracy of PG.

Our human-machine hybrid framework is complementary to research efforts on statistical PG models and spot-checking mechanisms of peer-prediction. The automated grader in our

framework can help to eliminate unreliable assignment scores to ensure that only quality scores are passed onto the statistical models such as the PG family model. Instead of setting up a spot-checking point in the peer grading, we make the automated grader automatically evaluate each submission, and calculate the error value of each peer score according to the score given by automated grader, and discard the peer scores with larger error value according to a reasonable threshold range. We give certain penalty to those who give these scores with high errors, such as subtracting their usual scores in this course, which can also present a certain incentive. In this way, the models can achieve better inference performance in aggregating peer scores. On the other hand, the automated grader can be used in spot-checking mechanisms and work as an online supervisor to perform verification tasks on behalf of TAs and update TAs with the screening results. With the assistance of the automated grader, TAs do not have to randomly choose among evaluations and only need to examine those evaluations that may be misjudged by the automated grader.

The development of a reliable automated grader is regarded as a challenging task. Recently, automatic essay grading has become a hot topic in the field of natural language processing. Many researchers such as [9], [10], [18] designed neural network-based automated graders to evaluate open essays. We believe that these efforts are complementary to our human-machine hybrid framework. The state-of-the-art automated graders cannot complete grading tasks in a fully autonomous way, especially for science essays and technical reports in domain-oriented courses. Thus, our framework assumes only an automated grader with limited classification capability and regards the grader as an intelligent assistant who can work with course instructors and TAs in the process of PG. If an advanced neural automated grader is used in our framework, this automated grader can certainly provide higher accuracy in eliminating unreliable peer scores and improve the overall performance of PG.

III. PROBLEM ANALYSIS OF PEER-GRADING MODELS

In this section, we first evaluate the PG models, then discuss the problems of the PG models when these models are applied in the SPOC settings. Through simulation experiments, we analyze the fault tolerance of the PG models with the increases in the number of undutiful students. We also compare the simulation results with the actual dataset collected from our SPOC experiments.

A. PG MODELS

We adopt the PG models [1]–[3], a Bayesian graph model with the latent factors including the biases and reliabilities of the peer graders, in SPOC scenarios. These models consider the dependency between the true score of a grader's submission and the grader's reliability and describe the reliability using common distributions such as Gaussian or Gamma distributions. In our previous research [3], we investigated the potential factors that may affect a grader's reliability except

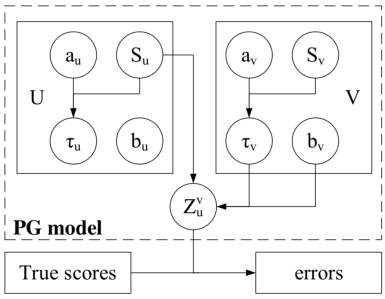

FIGURE 1. Bayesian graph of the peer-grading model. The dotted box represents the peer-grading model, in which the symbol v denotes a specific student grader, u denotes a specific student, a_v denotes the ability of the grader, s_v denotes the true score of the grader's submission, and τ_v and b_v denote the grader's reliability and bias, respectively. Assistant scores are generated by our course assistants, and can be regarded as the ground truth of every submission. The grading errors are the discrepancy between the assistant scores and the observation scores z_u^v given by the graders.

the graders' submission scores. We can use these factors to infer a grader's reliability and further to predict the score of a learner's submission.

Figure 1 shows that the observation score z_u^v is affected by latent factors including b_v, τ_v, and the learner's true score s_u. To evaluate the accuracy of the inferred true scores, we consider the TA scores as the ground truth, and the errors are the difference between the inferred true scores and the TA scores.

$$\tau_v \sim \mathcal{N}(\rho, 1/\beta_0) \tag{1}$$
$$b_v \sim \mathcal{N}(0, 1/\eta_0) \tag{2}$$
$$s_u \sim \mathcal{N}(\mu_0, 1/\gamma_0) \tag{3}$$
$$z_u^v \sim \mathcal{N}(s_u + b_v, \lambda/\tau_v) \tag{4}$$

Eqs (1)(2)(3)(4) show that the distributions of the major variables in the PG model are interrelated. Therefore, it is hard to make an exact inference from the PG model; thus, we must use approximate inference methods such as Gibbs sampling to fit the model. We also need to presuppose certain hyperparameter distributions, for example, a grader's bias and his or her submission's true score, as the prior distribution to train the model. Furthermore, we use the root mean square error (RMSE) to estimate the model. The output of the PG model is the prediction scores of the submissions.

B. LIMITATIONS OF PG MODELS IN SPOCs

There are two major factors that may prohibit SPOC students from performing PG tasks in a fair and accurate way. First, students without the right knowledge and dedication may regard a PG task as an unnecessary burdens and decide to give the assignments random scores. Second, the affinity among SPOC students who often interact with each other at the same campus or even classroom may drive these students to assign higher scores to their peers' submissions. Both factors may result in high deviations between the observed scores z_u^v and the ground-truth score. When many peer-graded scores with high errors occur, these errors can significantly affect the

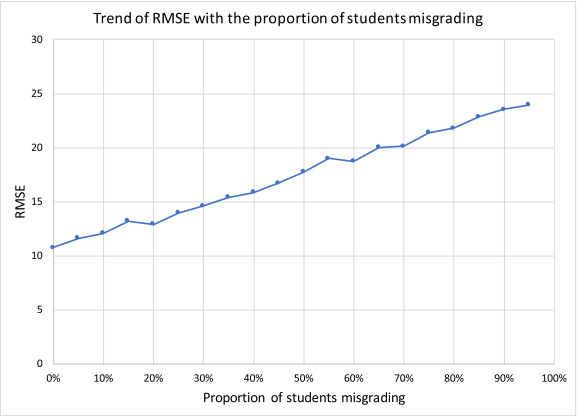

FIGURE 2. Correlation between the RMSE of the PG model and the proportion of undutiful students in the simulation.

inference performance of the PG models when we attempt to infer the latent true score s_u for each submission.

We run a simulation experiment to evaluate the impact of a student's attitude on peer assessment. In the experiment, student graders are divided into two groups: dutiful students with a good attitude toward grading and undutiful students who randomly give grading scores. Apparently, an increase in the number of undutiful students generates more data noise in the process of PG. By examining the correlation between the RMSE and the proportion of undutiful students, we can determine the tolerance of the PG models against data error generated by student graders.

Based on the configuration of the simulation, we extend the PG models as follows: Define the number of students with an undutiful attitude toward PG as p. The value of p gradually increases from 0 to n, where n is the total number of the students. Define a score distribution set D, and d_iD denotes a specific distribution that a student may choose to follow. The set D contains the distributions (5) and (6):

$$z_u^v \sim \mathcal{N}(s_u + b_v, \lambda/\tau_v) \tag{5}$$

$$z_u^v = x \pm random(y) \tag{6}$$

Eq (5) represents a strategy distribution in which the observed scores are presented by good students with a dutiful grading attitude, and Eq (6) represents the score distribution in which the observed scores are presented by undutiful students with high deviation. For simplicity, we assume that a student determines his or her choice of grading strategy before he or she accepts the grading task and does not change attitude in the middle of the grading process.

Figure 2 shows that the RMSE of the prediction scores is linearly correlated with the proportion of undutiful peer graders and the value of the RMSE is in the range [10], [25]. This result remains the same even when we change the parameters (x, y) in Eq (6). By analyzing the model of Eqs (1)(2)(3)(4), we conclude that in SPOCs, the RMSE is affected mainly by the number of undutiful students.

We further analyze the correlation through Eq (7). The expression of RMSE can be defined in Eq (7), where the set X_{model} denotes the prediction results of the PG model, $X_{model,k}$ denotes the specific prediction score assigned by a grader, the set X_{true} denotes the ground truth given by the TA and $X_{true,k}$ denotes the corresponding ground truth of one submission.

$$RMSE = \sqrt{\frac{\sum_{k=1}^{n}(X_{model,k} - X_{true,k})^2}{n}} \tag{7}$$

Suppose there are p submissions with undutiful peer scores. We can expand Eq (7) by separating the errors generated by the dutiful group and undutiful group. We define $e_k = X_{model,k} - X_{true,k}, k \in [1, n]$, and according to the mean inequality, we can obtain the following inequality relationships of the RMSE.

$$\begin{aligned} RMSE &= \sqrt{\frac{\sum_{i=1}^{p} {e_i}^2 + \sum_{j=p+1}^{n} {e_j}^2}{n}} \\ &\geq \sqrt{\frac{(\sum_{i=1}^{p} e_i)^2}{np} + \frac{(\sum_{j=p+1}^{n} e_j)^2}{n(n-p)}} \end{aligned} \tag{8}$$

We define $p\bar{e} = \sum_{i=1}^{p} e_i$ as the average of the set $A = \{e_i | i \in [1, p]\}$, and $(n-p)\bar{f} = \sum_{j=p+1}^{n} e_j$ denotes the average of the set $B = \{e_j | j \in [p+1, n]\}$. The condition for which the equality is satisfied is that each element in the set A is equal and each element in the set B is equal. Thus, we can transform Eq (8) into the following,

$$RMSE \cong \sqrt{\frac{p(\bar{e}^2 - \bar{f}^2)}{n} + \bar{f}^2} \tag{9}$$

Because of the assumption $|\bar{e}| \geq |\bar{f}|$, the value of the RMSE increases as p changes from 0 to n. Thus, in summary that the grading attitude of the students can significantly affect the performance of the PG model.

The impact on the RMSE of the students' grading attitude was found in the real peer-assessment experiment in the SPOC setting. In our previous research [3], the actual dataset demonstrates that the RMSE of the scores predicted by the PG models stays in the range [17], [19]. This high value of RMSE seems to be at the same level as the simulation results in Figure 2, when the ratio of the students with undutiful attitude is as high as 50%-60%. This observation confirms that we need to tackle this problem in SPOCs to achieve a low RMSE for the PG models.

C. COMPARISON AMONG GRADING ERROR DISTRIBUTIONS IN THE SIMULATION AND ACTUAL DATASETS

By comparing different inference performances of the PG models both in the simulation experiments and the real dataset, we analyze the effect of the features of the bias b_v and reliability τ_v on the precision of inferring the true score s_u. In the Gibbs sampling process for fitting the PG models,

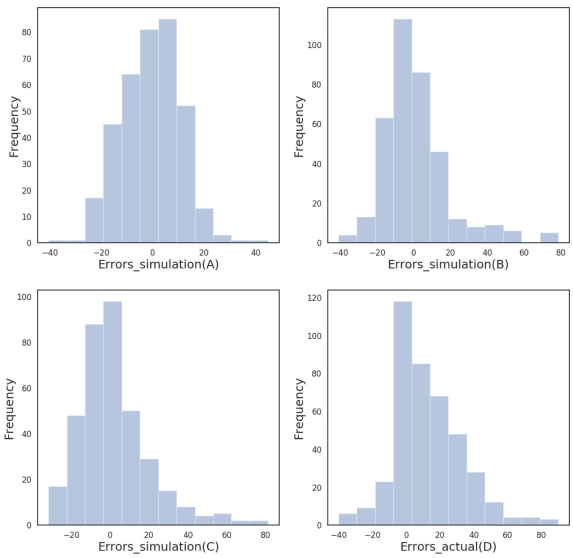

FIGURE 3. Distributions of errors in three simulation datasets and the actual dataset. Figure 3A plots a histogram of grading errors generated by the first simulation experiment without undutiful students. Figure 3B plots a histogram of grading errors generated by the second simulation experiment with 40% undutiful students. Figure 3C plots the histogram of grading errors generated by the third simulation experiment with 60% undutiful students. Figure 3D plots a histogram of grading errors collected from the Computer Network Experiments tutoring system.

Eq (10) updates s_u in constant iterations.

$$s_u \sim \mathcal{N}\left(\frac{\gamma_0\mu_0 + \beta_0\tau_{u_i} + \sum_{v:v\to u_i}\frac{\tau_v(z_u^v - b_v)}{\lambda}}{\gamma_0 + \beta_0 + \sum_{v:v\to u_i}\frac{\tau_v}{\lambda}}, \frac{1}{\gamma_0 + \beta_0 + \sum_{v:v\to u_i}\frac{\tau_v}{\lambda}}\right) \quad (10)$$

In Eq (10), the variable z_u^v is a constant value; in addition to τ_v and b_v, the other parameters are hyperparameters, and thus, the main factors affecting the true score s_u are a grader's bias and reliability. If the students randomly give a submission a score, there is no way to measure the reliability and bias, which seriously reduces the effectiveness of the PG model.

The real dataset is collected from the Computer Network Experiments tutoring system, with a total 724 students. As a comparison, we generate the simulation dataset by using the PG models and analyze the error distributions of the peer scores.

Figure 3 shows that the simulated and actual datasets have a very different error distribution. As the first simulation assumes that every student's grading behavior follows the Gaussian model defined in Eq (5), the error distribution shown in Figure 3A stays within $[-1.2, 0.7]$. In contrast, Figure 3D demonstrates that the disagreement range of the SPOCs dataset is much larger than that of the first simulation dataset, which indicates that many students' grading behavior does not satisfy a Gaussian distribution in the real dataset. To further confirm this conclusion, we conducted two other simulation experiments, in which we set 40% undutiful students and 60% undutiful students to follow the random grading behavior defined in Eq (6). The results of the second and the third simulation experiments are shown in Figure 3B and Figure 3C, which demonstrate a similar error range to that of Figure 3D. We find many extreme scoring values such as -40 and 80 in Figure 3B, 3C and 3D. These observations suggest that students in the SPOC experiment tend to exhibit random grading behavior.

TABLE 1. Comparing the minimum and maximum RMSE in the simulation and SPOCs datasets.

Data	Simulation Dataset			Actual Dataset		
	Min	Max	AVG	Min	Max	AVG
RMSE	0.71	6.67	3.69	1.23	40.12	20.68
	0.78	6.78	3.76	1.76	49.23	25.50
	0.82	7.34	4.08	2.32	52.34	27.33
	0.91	7.89	4.40	2.12	75.56	38.84
	0.93	8.31	4.62	2.56	95.12	48.84
Average	0.83	7.40	4.11	2.00	62.48	32.23

Clearly, such a high deviation in the peer scores from the ground truth is the reason why the PG models cannot achieve a low RMSE. This deviation is mostly caused by individual students' undutiful grading attitudes. This latent factor introduces a strong noise in the grading data for the PG models and thus eventually leads to a poor inference performance of the models. The RMSE measurement from our experiments can further confirm the above hypothesis. The average RMSE between the predicted scores s_u and the ground truth reaches 20.10 in the real dataset, which is much higher than 4.02 – the RMSE value in the first simulated dataset. To make a closer comparison of the RMSE, we select the top five assignments with the minimum and maximum RMSE from both datasets and present the results in Table 1.

Table 1 shows that the minimum RMSE values of the simulated dataset are close to that of the SPOC dataset, which suggests that there are certain reasonable peer scores in the real dataset. However, the maximum RMSE values of the datasets are very different because the random grading behaviors in the real experiment are worse than those in the simulation scenario, which significantly increases the RMSE in the final aggregated results.

IV. RUBRICS OF PEER-GRADING IN COMPUTER NETWORKING LAB COURSE

In this section, we describe the assignment and rubrics for peer grading in the SPOC course of our research and further investigate the reasons why students do not seriously complete the peer assignments. This course is a hands-on-networking lab course that intends to give students opportunity to develop practical skills on designing, configurating and troubleshooting computer networks. In this SPOC course, students learn knowledge about computer network by watching course video material covering topics such as Ethernet, IP routing and TCP protocols. Afterwards, they conduct experiments in a real test-bed networking environment. In each experiment, every student needs to set up virtual local networks according to the requirement of lab assignment and submit their lab reports.

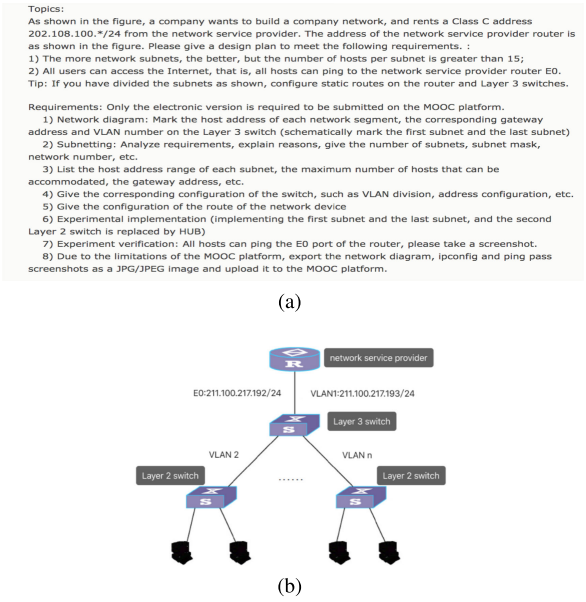

FIGURE 4. Subnetting assignments. Figure A describes the background and requirement of the assignment and lists six problems to be answered by the students. Figure B displays the diagram of network topology for this assignment.

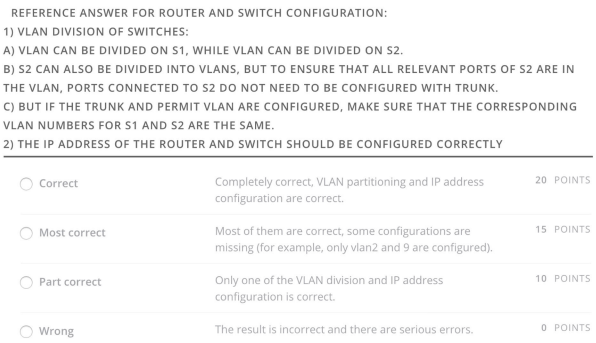

FIGURE 5. Categories for assigning score to a problem.

TABLE 2. Categories of each problem of the assignment.

Problem ID	Category 1	Category 2	Category 3	Category 4
1, 2, 3	0	5	10	—
4	0	10	20	—
5	0	10	15	20
6	0	10	—	—

Figure 4 shows an example of such an assignment on designing subnetworks in our course. Figure 4.A gives a description of the assignment that demands students to design a network for a company using a Class C address. Figure 4.B shows a diagram of the router connections, and students need to deploy the router in the actual physical network environment as required and add the routing configuration. The assignment consists of six problems including designing networking diagram, conceiving subnet division, configuring VLAN and IP routing and experimental verification. Each problem is designed to examine a student's mastery of specific skills in computer network lab. For example, Problem 2-3 checks whether a student can correctly divide a network into subnets and allocate host addresses in each subnet. Problem 4-5 mainly evaluates student's capability of configuring routers, switches and static routing.

In the process of PG, a student grader evaluates his peer's answer to these problems according to the rubrics specified by the course instructor. Figure 5 shows the rubric of the problem 4 about how to configure a router and switch in VLAN. This rubric presents four categories (Correct, Most Correct, Partially Correct, Wrong) for scoring the configuration plan in a laboratory report. This scoring rubric guides the students to determine the score for this answer based on his/her subjective judgement on the quality level of every answer.

Each rubric specifies categories and corresponding scores for the answers to the problem. Table 2 displays the categories of each problem. From the rubric design of the assignment problems, one can see that the number of rating categories of each question differs from two to four levels. The three-level rating category for Problem 1, 2 and 3 represents Correct, Partially Correct and Wrong. The major difference between Problem 4 and the other problems is that it needs more detailed rating categories to ensure more accurate rating results from graders. This problem involves more skills than other problems and thus needs more criteria to evaluate the quality level of the answers. The overall score of a lab report is a summation of all the score values of each problem.

As a summative rubric for peer grading, this rubric needs every student to compare others' submissions with reference solutions and determine their right quality level. Such a design and the inherent subject nature of peer grading leads to unavoidably high deviations between peer-graded scores and true scores. To cope with the major limitation of peer grading discussed in Section 3, it is necessary to combine an automated grading with peer grading. An automated grader should check the quality of each submission based on the rubric in order to filter random grading scores.

V. HUMAN-MACHINE HYBRID PEER-GRADING FRAMEWORK

This section presents the design of our human-machine hybrid framework in detail. Figure 6 displays the hybrid framework of human-machine PG that is designed to circumvent the problem caused by random grading behaviors in SPOC courses. The main idea of the framework is to use the scores estimated by the automated grader to screen the peer scores generated by undutiful students. The framework consists of three major components including a homework automated grader, a score filter and the PG models.

In the process of PG, the system first allocates the tasks for each student to perform their PG tasks. After the automated grader receives a score for a submission, the system estimates a score for the same submission and passes the estimation to the score filter. The score filter is responsible for comparing

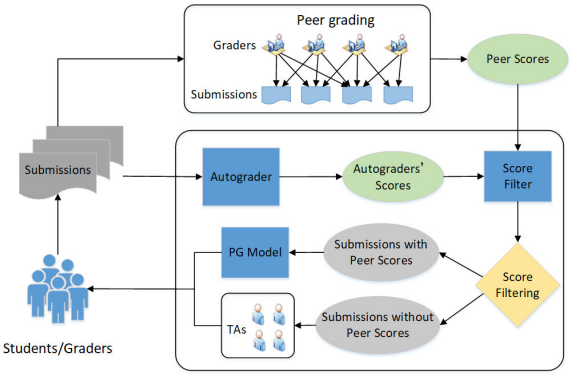

FIGURE 6. Human and machine hybrid framework of peer grading.

the estimate of the automated grader with the original peer score and abandoning the peer score if the deviation between these two scores goes beyond the predefined threshold. With the coordination of the automated grader and the score filter, the framework divides the student submissions into two groups: one group includes the submissions with legitimate peer scores that can be aggregated by the PG model for the score inference; the other group includes those without valid peer scores that have be to be sent to TAs for evaluation. In this way, we can alleviate the negative impact caused by the random grading behavior, thus improving the performance of the PG models in the SPOC setting. Most of the assignment submissions can still be handled by the PG models and only small fractions of the submissions need to be checked by TAs.

The automated grader can be designed based on traditional probabilistic models, such as naive Bayes, or deep learning methods, such as a convolutional neural network (CNN). By comparing the effectiveness of the automated graders designed by these ideas, we finally chose to use the CNN-based method. To illustrate the effectiveness of the CNN-based automated grader, we separately introduce two automated grading algorithms. One algorithm is to use a naive Bayesian model, and the other algorithm is based on a CNN classification model. As baseline algorithm in comparative experiments, we also designed a random filtering algorithm and KNN-based outlier detection algorithm to filter the original peer scores.

A. NAIVE BAYESIAN-BASED CLASSIFIER AS AUTOGRADER IMPLEMENTATION

We design a weak text classifier as an automated grader based on the naive Bayesian method in the hybrid PG framework. As discussed in Section 4, each lab report in our SPOC course often contains several problems, and thus, the design of the automated grader consists of several classifiers, each of which classifies one problem in the lab report. The score classification results for all the problems of the assignment are mapped into scores based on the rubric of the assignment and combined as the total score of the assignment report.

The specific design method of the automated grader classifier is as follows. Let $Y = \{Y : c_i \in Y\}$ to denote the set of the grading categories in a rubric for assessing student answers to a homework problem. For example, For example, the rubric of Problem 4 shown in Section 4 specifies four categories including: "Wrong", "Partly correct", "Mostly correct", and "Correct". For each grading category, there is a variable c_i, and $P(c_i)$ is the probability of selecting c_i.

1) Feature Selection: This classifier takes simple textural features in assignment reports. First, we build a set of correct answers to a specific problem and design a dictionary including 133 different keywords extracted from the report document as the features. In this process, words with high frequency and stop words are removed. Second, the automated grader counts the frequency of each feature from submission reports and obtains an integer vector $\vec{x}$ representing the frequency of the corresponding features in the dictionary for each submission report.
2) Naive Bayes Classifier: Let $P(\vec{x})$ be the probability of the joint feature distribution, $P(c_i|\vec{x})$ denote the posteriori probability that $\vec{x}$ belongs to a certain grading category c_i, and $P(\vec{x}|c_i)$ denote a condition probability that a feature $\vec{x}$ belongs to a category c_i.
3) The score estimation model is shown as Eq (11):

$$P(c_i|\vec{x}) = \frac{P(\vec{x}|c_i)P(c_i)}{P(\vec{x})} \tag{11}$$

The automated grader selects the grading category with the maximum probability $P(c_g|\vec{x})$ according to Eq (12).

$$P(c_g|\vec{x}) = \arg\max_{c_i} \frac{P(c_i)\prod_j P(x_j|c_i)}{\sum_i P(c_i)\prod_j P(x_j|c_i)} \tag{12}$$

$P(x_j|c_i)$ represents the probability that a feature belongs to a category. We assume that each feature should be independent. The conditional probability can be expanded into the product of the independent event probability, which simplifies the calculation.

We also introduce an assumption that keywords are independent of each other. In Eq (11), because $P(\vec{x})$ is a fixed value, we only need to compare the value of $P(\vec{x}|c_i)P(c_i)$. The Eq (12) can be simplified as follows.

$$P(c_i|\vec{x}) = \arg\max_{c_i} P(c_i)\prod_j P(x_j|c_i) \tag{13}$$

B. CNN-BASED CLASSIFIER AS THE AUTOMATED GRADER IMPLEMENTATION

Since CNN models are more capable of reflecting the logical order relationship of keywords in text than long short-term memory (LSTM) models, we adopt a CNN neural network in our automated grader algorithm. Figure 7 illustrates the diagram of the CNN-based classifier. The main processing pipeline includes a custom dictionary, the Jieba word segmentation method, word2vec for generating word vectors and a CNN network structure model.

The first processing step of the classifier pipeline is to run the Chinese word segmentation and Word2Vec to generate the right word embeddings by using historical submissions.

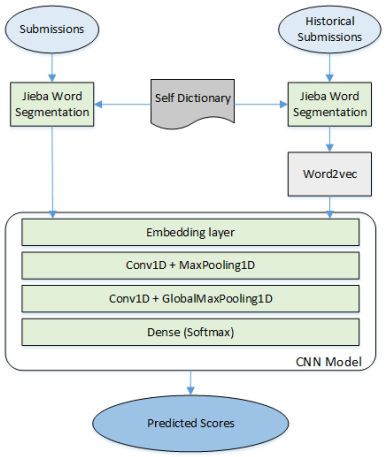

FIGURE 7. The CNN-based autograder model.

We create a customized dictionary containing the keywords in the rubrics for the assignments from the training dataset. Because the answers to the assignments are not plain text or pure numbers but a combination of text, IP addresses and network commands, using the pre-trained dictionary in the Jieba tool does not perform well when the pipeline builds the word vector for the text of lab reports. Therefore, we manually extract certain keywords so that these keywords are not separated, which improves the word segmentation results. In this paper, using word2vec to build a word vector can maintain the logical order of the keywords before and after, avoiding the higher scores for low-quality reports with only these keywords but poor grammar and content. These word vectors are further used as input data for the CNN model. We choose a three-layer CNN network structure, where the first and second layers are one-dimensional convolutional layers, and the last layer is a multicategory out layer.

C. METHOD OF OUTLIER DETECTION

To verify that the naive Bayes-based and CNN-based automated graders can improve the performance of the PG model in the human-machine hybrid framework, we select the outlier detection method as a baseline in comparison tests. Through the preliminary experiment, we compare the accuracy of three commonly used outlier detection methods: K-nearest neighbors (KNN), principal component analysis (PCA), and cluster-based local outlier factor (CBLOF). We find that the prediction accuracy of the PCA and CBLOF models is relatively high when the number of samples differs, and the performance of the KNN model is relatively stable. Therefore, KNN is chosen as the method of outlier detection.

As shown in Figure 8, there are three possible methods to apply KNN-based outlier detection for score filtering in the PG process: 1) All-Set Detection: to implement a global outlier detection by merging all the peer-graded scores of the submissions in a list to detect outliers; 2) Grader-Oriented Detection: to perform outlier detection on all the scores of lab reports reviewed by each rater; 3) Category-based Detection: to categorize the peer-graded scores into multiple groups according to the values, and performing outlier detection on the categories. For example, in this paper, we perform outlier detection on the peer scores with the three groups namely 60, 55 and 50.

For the three methods, we conducted the comparative experiments. In each experiment, we conducted and record the average value of 50 rounds of running the KNN model. We use the Euclidean distance formula based on the idea of five nearest neighbors. In the final experiment,we choose the Category-based Detection method. The specific experimental results are analyzed in Section 6.

D. SCORE FILTERING AND POSTPROCESSING

The score filter in the hybrid human-machine grading framework adopts a simple filtering process. This filter runs the following three steps: computing the absolute values of the differences between the scores estimated by the automated grader and the peer-graded scores, sorting these scores in a descending order, and filter out the top 20% with the highest deviation values. The design of the score filter involves two major issues: the threshold for dropping unreasonable scores and the postprocessing strategy for supplementing abandoned scores.

1) **Error Threshold of Score Filtering**
   Since our automated grader is a weak classifier, we need to consider its classification error during the automated grading process for evaluating the quality of all the problems in a lab report. We use the following equation to calculate the grading error.

$$Threshold_{error} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - a_i)^2}{n}} \tag{14}$$

   In Eq (14), $x_i \in \{x_1, x_2 \ldots, x_n\}$ denotes the score given by a student grader and $a_j \in \{a_1, a_2, \ldots, a_n\}$ denotes the score estimated by the automated grader. The value of n is the number of the problems in an assignment. We define Eq (14) to predict the error for each peer-graded score and sort the scores in descending order according to the value of the prediction error, thus filtering out the peer scores with high error values.

2) **Postprocess Strategy of Score Filtering**
   This simple algorithm may cause potential problems for the PG models. After the score filter drops these unreasonable peer-scored scores, this algorithm may create extreme cases where most peer scores for a student assignment are eliminated. In such a case, a postprocessing step is necessary in the score filter to supplement new scores for the downstream PG models.

For the postprocessing step, we propose the following three strategies to handle the scores filtered out.

1) Dropping-only Strategy: The score filter simply drops the scores identified by the automated grader and does not supplement any new scores.

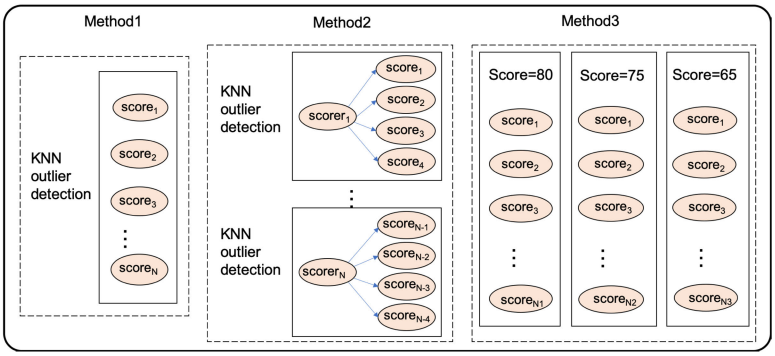

FIGURE 8. Classification method for outlier detection.

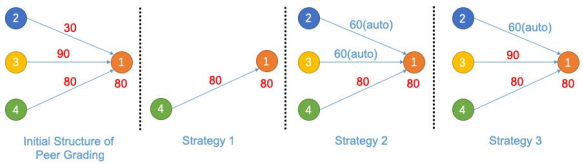

FIGURE 9. Three strategies of postprocessing to replace the filtered scores.

2) Replacement-by-Autograding Strategy: The score filter directly uses the scores generated by the automated grader to replace the peer scores that are identified as biased.
3) Mixed-Replacement Strategy: As a contrast strategy, the score filter can choose the replacement score among the peer scores and the score predicted by the automated grader based on the absolute difference from the ground truth. Although it is impossible to implement this strategy in the real system, this strategy gives us an upper bound for the strategy design when the ground-truth is available.

Figure 9 presents the examples of all the strategies. The leftmost graph panel depicts the relationship between the original peer scores and the real score of the same submission; from left to right, the second panel shows the score aggregation method using the Dropping-only strategy, the third panel illustrates the Replacement-by-Autograding strategy, and the last panel represents the use of the Mixed-Replacement strategy for score aggregation.

VI. EXPERIMENTAL RESULTS

We conduct experiments to evaluate the performance of the human-machine PG framework. We aim at evaluating whether the framework can effectively improve the accuracy of predicting scores for submissions. In this section, we first estimate the accuracy of the automated grader and verify which strategy is the most effective in the replacement of the filtered scores. Then, we study how to tune the threshold parameter of the score filter. Finally, we present the experimental results about the overall performance of the human-machine hybrid framework on three different datasets. The PG experiment was conducted on the computer network lab course that is offered to the senior college students pursuing a computer science major. Our experimental dataset was collected from the class sessions in 2015-2017, including a total of 6 PG assignments, 724 students and 2354 peer scores.

TABLE 3. Prediction accuracy of automated grader.

Problem ID	Naive Bayes	LSTM	CNN
1.Subnet Division	65.29%	76.56%	**85.34%**
2. Address Allocation	73.23%	78.32%	**86.54%**
3. Switch and Router Configuration	71.43%	81.45%	**82.76%**
4. Static Routing Configuration	61.45%	77.23%	**84.23%**

A. MODEL PERFORMANCE ANALYSIS

To evaluate the performance of the automated grading algorithms, we expect both the grading accuracy but also the reliability of the grading algorithm. We choose homework reports on the subnetwork chapter of the course as the training and test data to develop the three classifiers of the automated grader, namely the naive Bayes, LSTM, and CNN method. The data are collected from the 2015-2017 undergraduate and postgraduate subnetting experiments. After preprocessing the data, we use 5-fold cross validation to do the experiment based on the data of a total of 1,260 submissions.

1) Prediction accuracy of the automated grader
Table 3 displays the score prediction accuracy for the automated grader based on the naive Bayes, LSTM, and CNN method, respectively. The accuracy refers to the percentage of the automated grader predicted scores that are the same as the ground truth. From the experimental results, one can see that the CNN-based automated grader achieves the highest prediction accuracy. Thus, in this paper, we adopt the CNN-based algorithm as the automated grader in the human-machine hybrid framework.
2) Uncertainty analysis of the automated grader
Because the automated grader plays a critical role in the human-machine hybrid peer grading, we prefer the most reliable model with lowest predicative

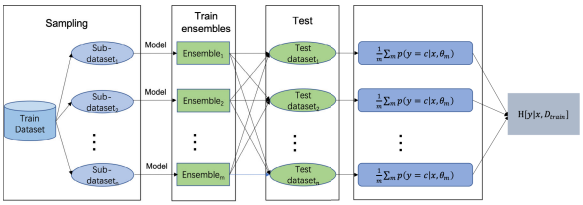

FIGURE 10. Model uncertainty experimental method.

uncertainty. Model uncertainty often represents the degree of confidence in predictive output of the model. We adopt an ensemble-based approach to quantify the uncertainty of our proposed models due to its simplicity, which is defined in Eq (14). This approach trains an ensemble of M classifiers with different subsets of the whole training dataset, and estimates the classification probability by running the averaged softmax vectors of each ensemble member [16].

$$H[y|x, D_{train}] := -\sum_{i=1}^{n}(\frac{1}{m}\sum_{m} p(y=c|x,\theta_m)) \cdot \log(\frac{1}{m}\sum mp(y=c|x,\theta_m)) \quad (15)$$

where y represents the classification result for x and θ_m denotes for the weight vectors in the ensemble model m, D_{train} denotes the training data set, c represents the correct category, n represents the number of test samples. Based on the average classification distribution on every data sample across all the ensemble models, we compute the model uncertainty in form of entropy. Figure 10 illustrates the entire computing workflow of uncertainty quantification method.

We extracted 80% of the data set as the training set and the remaining 20% as the test set. The uncertainty quantification algorithm firstly randomly samples the training set, extracts 100 subsets from the train data set, and builds 100 ensembles on the training set. To calculate the uncertainty of the model, we first calculate the cross-entropy of each sample in the test set. During the testing, we take a sample from the test set, run the 100 trained ensembles to generate their prediction results, and use cross entropy to calculate the sample's uncertainty for the model. After this uncertainty quantification process gradually run the above measurement step for all the test data samples, it can calculate the average value of the uncertainty across all data as the uncertainty value of the automated grader. We experimented with the naive Bayesian and CNN-based automated graders. Figure 11 plots the experiment results of the Bayes and CNN's prediction accuracy in term of variance from ground-truth scores at each ensemble model. As can be seen from the figure, not only the accuracy of the CNN model seems better than Bayes, but its fluctuation range is also smaller than that of Bayes.

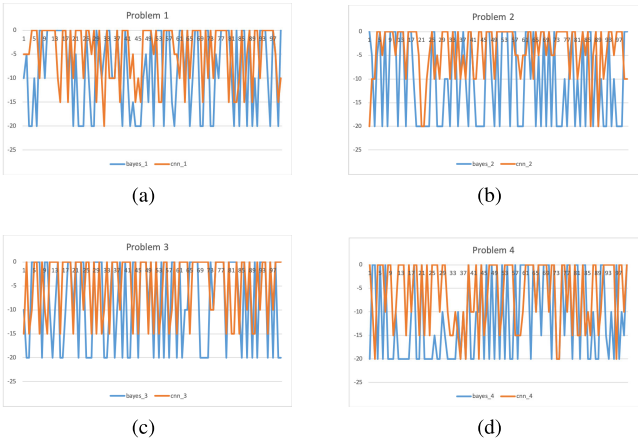

FIGURE 11. Score Prediction Variance of Model ensembles. Each value on the horizontal axis represents an ensemble, and the value on the vertical axis represents the difference between the predicted value of each ensemble and its corresponding groundtruth, and problem 1-4 respectively represent the problems in table 2.

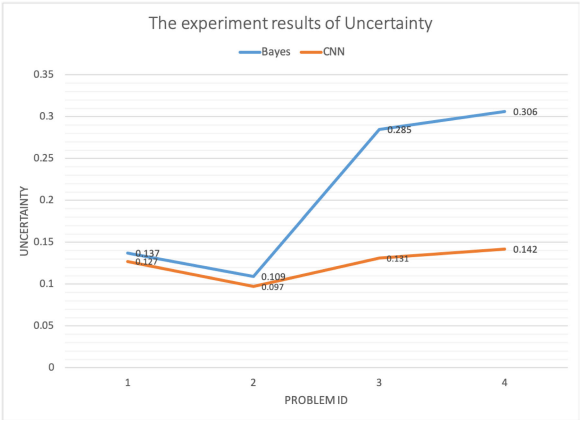

FIGURE 12. Model Uncertainty of CNN and naive Bayesian methods.

Figure 12 compares the uncertainty of the Naive Bayes model with that of the CNN model. It shows that when using the CNN model to predict peer scores, the uncertain results in all the problems are smaller than using the Naive Bayes result. In other words, the results are more stable in case of using the CNN model to predict the peer scores. Therefore, given the better reliability of the CNN model, it make sense to choose the CNN-based automated grader as the score filter in the framework.

B. CHOICE OF POSTPROCESSING STRATEGIES FOR SCORE FILTERING

We evaluate the performance of the score filter, especially the postprocessing strategy. In addition to the three strategies described in Section V, we run the postprocessing with the ground-truth strategy, in which the filtered top 20% peer scores are replaced by the ground-truth value. We use 5-fold cross validation to calculate RMSE under different strategies. Table 4 shows that the dropping-only strategy achieves better performance than the Replacement-by-Autograding

TABLE 4. RMSE of the three postprocessing strategies.

Postprocessing Strategy	Dropping Only	Replacement by Automated grading	Mixed Replacement
RMSE	14.78	16.43	**13.21**

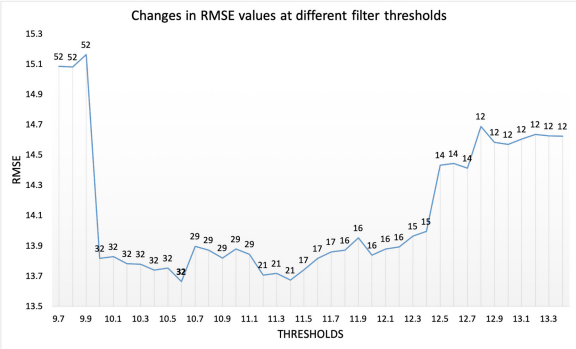

FIGURE 13. Changes in RMSE in the PG model with different thresholds.

TABLE 5. Experimental results of outlier detection.

		All-Set Dectection	Grader-Oriented Detection	Category-based Detection
RMSE	PG1	22.21	20.88	21.43
	PG3	23..22	22.76	21.11
	PG4	24.44	22.57	22.09
	PG5	22.45	21.85	20.35

TABLE 6. Experimental results of using human-machine frameworks.

	RMSE				
Models	Original PG Models Without Score Filtering	With Random Filtering	Outlier Detection	With Naive Bayesian-based Automated Grader Filtering	With CNN-based Automated Grader
PG1	21.34	21.90	21.05	16.89	**12.54**
PG3	20.12	22.02	21.10	16.31	**11.43**
PG4	21.90	22.43	22.30	17.20	**11.32**
PG5	21.65	21.67	20.50	17.30	**12.20**

strategy, possibly due to the limited grading accuracy of the classifier in the automated grader. Although both the mixed-Replacement strategy and the ground-truth strategy achieve the lowest RMSE, their implementations of these methods are not feasible in actual scenarios. Therefore, we choose the Dropping-only strategy for postprocessing in the score filter. The strategy requires that if all the peer scores of a submission are eliminated, the submission must be sent to the TAs for assessment.

C. TUNING THE THRESHOLD OF THE SCORE FILTER

The error threshold may play a vital role in the PG framework because this parameter can determine whether a peer score should be abandoned or be re-evaluated by class TAs. We investigated the impact of the error threshold by comparing the RMSEs generated by the PG models and the automated grader under different threshold values by using 5-fold cross validation experiment.

Figure 13 shows that the RMSE calculated by the PG model decreases to the smallest value when the threshold of the score filter is set to 10.6, and that the number of submissions without any peer scores is 32; Therefore, after automated grader filtering, the number of submissions to be sent to the TAs for review does not add an additional burden to the TAs. This paper chooses the threshold value of 10.6 in the following experiments.

D. SELECTION OF OUTLIER DETECTION METHODS

In order to use the KNN-based method to effectively detect outliers in the dataset, we proposed three different data grouping methods for the data set in Section 5. Here, based on these three data groupings, different PG models are used to calculate the process and RMSE is used as a comparison standard for different models. We use 5-fold cross validation to calculate RMSE under different detection methods. It can be seen from Table 5 that the experimental results of All-Set Detection are significantly weaker than Grader-Oriented Detection and Category-based Detection. In Category-based Detection, although the RMSE calculated by the PG1 and PG4 models is litter higher than Category-based Detection, the RMSE of the two is not much different in PG4, and the RMSE calculated by the PG5 model is significantly lower than the value of Grader-Oriented Detection. Therefore, we chose Category-based Detection as our data grouping method in the final comparison experiment.

E. OVERALL PERFORMANCE OF THE HYBRID PEER-GRADING FRAMEWORK

To evaluate the performance of the hybrid PG framework, we run the PG models after the peer-scored scores are filtered by either the framework or random filtering. In this way, we can generate five groups of experimental data: an initial dataset without any score filtering, a dataset with naive Bayesian-based automated grader filtering, a dataset with random filtering, a dataset with outlier detection, and a dataset with the CNN-based automated grader. We use 5-fold cross validation to calculate the RMSE value of the PG model under different filtering methods. The RMSEs of the PG models on all the data sets are shown in Table 6.

After the peer scores are sorted in descending order of the estimated error, the top 20% of the scores are filtered out in each experiment. The filter process may eliminate all the peer scores for certain submissions, which have been re-evaluated by the class TAs. In the above experiment, when the error threshold is set to 10.6 with the CNN-based automated grader, 689 submissions are left with at least one peer score.

The training process of the PG models is performed through the Hamilton sampling method implemented in the tool of Stan [11]. The final RMSE is calculated by the average of 500 iterations. Table 6 shows that regardless of the PG model used, the human-machine framework can obtain

the best performance, which reduces the RMSE by 8-9 on average from theoriginal solution without score filtering. This outcome confirms that the hybrid human-machine PG framework can improve the prediction accuracy of the PG models with the presence of random grading behavior.

VII. CONCLUSION AND FUTURE WORK

In this paper, we introduce a novel human-machine hybrid PG framework to alleviate the problem of random grading, where student graders perform their PG tasks in an undutiful manner. The most important component of the framework is an automated grader that can classify students' submissions using machine learning and enable the framework to filter out the peer scores with high errors. When filtering the peer scores, the framework calculates the error threshold according to the RMSE metric. Extensive experiments confirm that the hybrid framework can effectively eliminate the noise in peer scores made by undutiful student graders and improve the prediction accuracy of the PG models.

As a preliminary work, the hybrid PG framework can be improved in the following aspects. At the current implementation, the score filter of the hybrid framework only plays the role of screening extremely deviated peer-graded scores but cannot regulate the grading behavior of students. We plan to take a game theoretic approach and design a competition mechanism to correct every student's grading attitude and motivate the students to provide more accurate assessments of their peers' submissions.

REFERENCES

[1] C. Piech, J. Huang, Z. Chen, C. Do, A. Ng, and D. Koller, "Tuned models of peer assessment in MOOCs," 2013, *arXiv:1307.2579*. [Online]. Available: http://arxiv.org/abs/1307.2579

[2] F. Mi and D. Y. Yeung, "Probabilistic graphical models for boosting cardinal and ordinal peer grading in MOOCs," in *Proc. AAAI 29th AAAI Conf. Artif. Intell.*, 2015, pp. 454–460.

[3] Y. Han, W. Wu, and X. Zhou, "Improving models of peer grading in SPOC," in *Proc. EDM*, 2017, pp. 1–2.

[4] M. Uto and M. Ueno, "Item response theory for peer assessment," *IEEE Trans. Learn. Technol.*, vol. 9, no. 2, pp. 157–170, Apr./Jun. 2016, doi: 10.1109/TLT.2015.2476806.

[5] X. A. Gao, A. Mao, Y. Chen, and R. P. Adams, "Trick or treat: Putting peer prediction to the test," in *Proc. 15th ACM Conf. Econ. Comput. (EC)*, 2014, pp. 507–524, doi: 10.1145/2600057.2602865.

[6] Y. Song, Z. Hu, and E. F. Gehringer, "Collusion in educational peer assessment: How much do we need to worry about it?" in *Proc. IEEE Frontiers Edu. Conf. (FIE)*, Indianapolis, IN, USA, Oct. 2017, pp. 1–8, doi: 10.1109/FIE.2017.8190621.

[7] A. Luca, P. Vassilis, and S. Michael, "Incentives for truthful peer grading," Dept. Comput. Sci. Game Theory, Univ. California Santa Cruz, Santa Cruz, CA, USA, Tech. Rep., 2016. [Online]. Available: https://arxiv.org/abs/1604.03178

[8] A. Gao, J. R. Wright, and K. Leyton-Brown, "Incentivizing evaluation via limited access to ground truth: Peer-prediction makes things worse," 2016, *arXiv:1606.07042*. [Online]. Available: http://arxiv.org/abs/1606.07042

[9] K. Taghipour and H. T. Ng, "A neural approach to automated essay scoring," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1882–1891, doi: 10.18653/v1/D16-1193.

[10] M. Nitin and C. Aoife, "Automated scoring: Beyond natural language processing," in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 1099–1109.

[11] C. Bob, G. Andrew, L. Daniel, and G. Jiqiang, "Stan: A probabilistic programming language for Bayesian inference and optimization," *J. Educ. Behav. Statist.*, vol. 40, no. 5, 2015, pp. 530–543, doi: 10.18637/jss.v076.i01.

[12] W. Wu, C. Daskalakis, N. Kaashoek, C. Tzamos, and M. Weinberg, "Game theory based peer grading mechanisms for MOOCs," in *Proc. 2nd ACM Conf. Learn. @ Scale (L@S)*, 2015, pp. 281–286, doi: 10.1145/2724660.2728676.

[13] A. Dasgupta and A. Ghosh, "Crowdsourced judgement elicitation with endogenous proficiency," in *Proc. 22nd Int. Conf. World Wide Web (WWW)*, 2013, pp. 319–330, doi: 10.1145/2488388.2488417.

[14] V. Shnayder, R. Frongillo, and A. Agarwal, "Strong truthfulness in multi-task peer prediction," Tech. Rep., 2016.

[15] W. Wang, B. An, and Y. Jiang, "Optimal spot-checking for improving evaluation accuracy of peer grading systems," in *Proc. 32nd AAAI Conf. Artif. Intell.* Menlo Park, CA, USA: AAAI, 2018, pp. 833–840.

[16] W. H. Beluch, T. Genewein, A. Nurnberger, and J. M. Kohler, "The power of ensembles for active learning in image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, Jun. 2018, pp. 9368–9377, doi: 10.1109/CVPR.2018.00976.

[17] W. Wang, B. An, and Y. Jiang, "Optimal spot-checking for improving the evaluation quality of crowdsourcing: Application to peer grading systems," *IEEE Trans. Comput. Social Syst.*, vol. 7, no. 4, pp. 940–955, Aug. 2020, doi: 10.1109/TCSS.2020.2998732.

[18] K. Zupanc and Z. Bosnić, "Improvement of automated essay grading by grouping similar graders," *Fundam. Informaticae*, vol. 172, no. 3, pp. 239–259, 2020.

YONG HAN was born in Zhoukou, Henan, in 1988. He received the master's degree from the University of Science and Technology Beijing, in 2012. He is currently pursuing the Ph.D. degree with the Beihang University. He studied under Professor Li Wei and the Instructor Professor Wu Wenjun. His main research interests include analysis of student behavior in pedagogy, homework evaluation, including mutual score aggregation and game theory-based learning incentive mechanism.

WENJUN WU is currently a Professor with Beihang University, a Ph.D. Tutor, the Deputy Director of the State Key Laboratory of Software Development Environment, and the Deputy Head of the National Artificial Intelligence Standards Group. He has long studied the service software theory and service system in the field of scientific big data and computing. He studied and participated in many important sciences of the US NSF during his studies and worked with Indiana University and the University of Chicago Agang National Laboratory, from 2002 to 2010. His service platform research projects, including computing social and behavioral science big data analysis service platform, open life science knowledge computing grid, next generation high-throughput gene sequencing grid, multimedia real-time interactive collaborative service environment, proposed multimedia collaborative services and based on social network-based scientific service gateway and other innovative technologies, the scientific service software platform developed by the company is widely used by more than 100 scientific research institutions in the world. In recent years, in the domestic research group software development method and cloud platform-based service software technology, hosted and participated in the National Natural Science Foundation, the National 863, the National 973, and other topics in the service computing and distributed systems conference IEEE SC, ICWS more than 100 academic papers, such as IPDPS, and international academic conferences, such as the IEEE Internet Computing and Simulation Modelling in Q1 area, systematically proposed the theoretical framework and process model of group software, and published the theory of group software for the first time in the world.

YITAO YAN was born in Changzhi, Shanxi, in January 1996. He received the bachelor's degree from Tianjin University, in 2018. He is currently pursuing the degree with the State Key Laboratory of Software Development Environment, Beihang University. His instructor is Professor Wu Wenjun. His research interest includes learning incentive mechanism-based on game theory. He has received the National Innovation Award of Dacheng and the Sanhao Student of Tianjin University.

LIJUN ZHANG was born in 1971. He received the degree in control theory and application from Beihang University, in 1998. He was named as an Associate Professor, in 2001. He is currently an Associate Professor and the Deputy Director of the Teaching Experiment Center, School of Computer Science, Beihang University. More than 30 master students have been instructed. As the project leader, he completed an aviation science fund and participated in the research work of several national natural science funds and aviation science funds. He is responsible for undergraduate and graduate students in the computer network experiment course. He has published more than 20 academic papers in domestic and international publications and conferences. His research interests include computer network technology, control theory and application, complex network theory and application, and computer application project development. He has received the second prize of national teaching achievement, the first prize of Beijing teaching achievement, and the second prize of scientific and technological progress of the National Defense Science and Technology Commission.