Learning Automated Essay Scoring Models Using Item-Response-Theory-Based Scores to Decrease Effects of Rater Biases

Masaki Uto and Masashi Okano

***Abstract*—In automated essay scoring (AES), scores are automatically assigned to essays as an alternative to grading by humans. Traditional AES typically relies on handcrafted features, whereas recent studies have proposed AES models based on deep neural networks to obviate the need for feature engineering. Those AES models generally require training on a large dataset of graded essays. However, assigned grades in such a training dataset are known to be biased owing to effects of rater characteristics when grading is conducted by assigning a few raters in a rater set to each essay. Performance of AES models drops when such biased data are used for model training. Researchers in the fields of educational and psychological measurement have recently proposed item response theory (IRT) models that can estimate essay scores while considering effects of rater biases. This study, therefore, proposes a new method that trains AES models using IRT-based scores for dealing with rater bias within training data.**

***Index Terms*—Automated essay scoring (AES), deep neural networks (DNNs), item response theory (IRT), rater bias.**

I. Introduction

In various assessment fields, essay writing tests have attracted much attention as a way to measure practical and higher order abilities, such as logical thinking, critical reasoning, and creative thinking [1]–[5]. In essay writing tests, examinees write essays about a given topic, and human raters grade those essays based on a scoring rubric. However, grading can be an expensive and time-consuming process when there are many examinees [2], [6]. In addition, grading by humans is not always sufficiently accurate even when a rubric is used because assigned scores depend strongly on rater characteristics, such as strictness and consistency [7]–[14]. Automated essay scoring (AES), which utilizes natural language processing (NLP) and machine learning techniques to automatically grade essays, is one approach toward resolving this problem.

Many AES methods have been developed over recent decades, and these can generally be classified as *feature-engineering* or *automatic feature extraction* approaches [2], [6]. The feature-engineering approach predicts scores using manually tuned features, such as essay length and number of spelling errors (see, e.g., [15]–[18]). The advantages of this approach include interpretability and explainability. However, this approach generally requires extensive effort in engineering effective features to achieve high scoring accuracy for various datasets. To obviate the need for feature engineering, automatic feature extraction approaches based on deep neural networks (DNNs) have recently attracted attention. DNNs, which have recently achieved tremendous success in various domains, are a type of machine learning model composed of multiple neural networks designed to mimic the behavior of the human brain. Many DNN-AES models have been proposed in the past few years and have achieved state-of-the-art accuracy (see, e.g., [19]–[30]).

Those AES models generally require a large dataset of essays graded by human raters as training data. When creating a training dataset, essay grading tasks are generally shared among many raters by assigning a few raters to each essay to lower assessment burdens. However, in such cases, assigned scores are known to be biased owing to the effects of rater characteristics [9]–[12], [14], [31]–[35]. Performance of AES models drops when biased data are used for model training because the resulting model reflects the bias effects [15], [36]–[38]. This problem has been generally overlooked or ignored, but it is a significant issue affecting all AES models that use supervised machine learning models, including DNNs. Furthermore, in practice, it is generally difficult to create a training dataset without rater bias effects because to do so would incur high costs for rater training and data quality confirmation.

In the fields of educational and psychological measurement, statistical models for estimating essay scores while considering rater bias effects have recently been proposed. Specifically, they are formulated as item response theory (IRT) models that incorporate parameters representing rater characteristics [7], [13], [39]–[43]. Such models have been applied to various performance tests, including essay writing tests. Previous studies have reported that they can provide accurate scores by removing adverse effects of rater bias (see, e.g., [32] and [41]–[44]).

This study, therefore, proposes a new method that trains AES models using IRT-based scores for dealing with rater bias in training data. In our method, an IRT model is first

Manuscript received May 14, 2021; revised November 6, 2021; accepted January 16, 2022. Date of publication January 25, 2022; date of current version April 13, 2022. This work was supported by the Japan Society for the Promotion of Science through the Grants-in-Aid for Scientific Research Program under Grant 19H05663 and Grant 20K20817. This article is a revised and expanded version of a paper that was presented at the 21st International Conference on Artificial Intelligence in Education, Ifrane, Morocco, Jul. 2020 [DOI:10.1007/978-3-030-52237-7_44]. *(Corresponding author: Masaki Uto.)*

The authors are with the Graduate School of Informatics and Engineering, University of Electro-Communications, Chofu 182-8585, Japan (e-mail: uto@ai.lab.uec.ac.jp; okano@ai.lab.uec.ac.jp).

Digital Object Identifier 10.1109/TLT.2022.3145352

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

applied to raw rating data to estimate scores that remove effects of rater bias. Then, an AES model is trained using the IRT-based scores. Because the IRT-based scores are theoretically free from rater bias effects, the AES model will not reflect the bias effects. Our method is simple and easily applied to various conventional AES models, and our experimental results show that it effectively improves AES performance. Moreover, this method is highly suited to educational contexts and to low- and medium-stake tests because preparing high-quality training data in such situations is generally difficult, owing to cost concerns.

Note that Aomi *et al.* [45] proposed another AES method involving IRT. However, unlike our method, their method does not address the problem of rater biases in training data because their objective was to integrate the predicted scores of multiple AES models to improve scoring accuracy.

II. Data

We assume that a training dataset consists of essays written by J examinees and essay rating data assigned by R raters. Let e_j be an essay by examinee $j \in \mathcal{J} = \{1, \ldots, J\}$ and let $U_{jr} \in \mathcal{K} = \{1, \ldots, K\}$ represent a categorical rating assigned by rater $r \in \mathcal{R} = \{1, \ldots, R\}$ to e_j. The rating data can then be defined as

$$U = \{U_{jr} \in \mathcal{K} \cup \{-1\} \,|\, j \in \mathcal{J}, r \in \mathcal{R}\} \tag{1}$$

with $U_{jr} = -1$ denoting missing data. Missing rating data occur because only a few raters in $\mathcal{R}$ can practically grade each essay e_j to reduce assessment workload.

Furthermore, letting $\mathcal{V}$ be a vocabulary list for essay collection $\boldsymbol{E} = \{e_j \,|\, j \in \mathcal{J}\}$, essay $e_j \in \boldsymbol{E}$ is definable as a sequence of vocabulary words

$$e_j = \{w_{jt} \in \mathcal{V} \,|\, t = \{1, \ldots, n_j\}\} \tag{2}$$

where w_{jt} is the tth word in e_j and n_j is the number of words in e_j.

This study is aimed at training AES models using these data.

III. AES Models

This section presents a review of conventional AES models based on the feature-engineering approach and the automatic feature extraction approach.

A. Feature-Engineering Approach

The feature-engineering approach predicts scores using textual features, which are manually designed by human experts. Typical features are essay length and number of grammatical and spelling errors. This approach first calculates such textual features from a target essay text and, then, typically inputs the feature vector to a regression model and outputs a score.

This approach has long been used in various AES models (see, e.g., [18] and [46]–[49]). E-rater [46], which has been developed and used by Educational Testing Service, is a representative feature-engineering approach model based on a linear regression model. Another recent popular feature-engineering approach model is the Enhanced AI Scoring Engine (EASE) [47], which achieved high performance in the Automated Student Assessment Prize (ASAP) competition on Kaggle. EASE uses Bayesian linear ridge regression with several feature types, including length-based features, part-of-speech-based features, prompt-relevant features, and bag-of-words-based features.

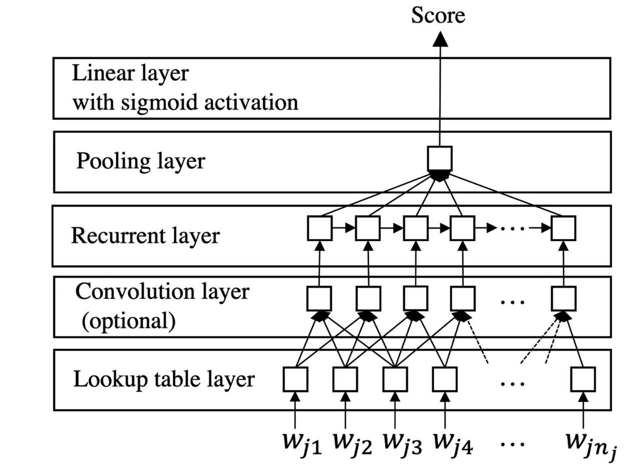


Fig. 1. RNN-based model architecture.

Feature-engineering approach models generally require training of regression models using a training dataset, although e-rater uses empirically determined weights for the regression model.

B. Automatic Feature Extraction Approach

As models based on the automatic feature extraction approach, DNN-based AES models have attracted wide attention. Although many DNN-based models have been proposed in the past few years (see, e.g., [19]–[29]), we introduce the most popular model, which is based mainly on a recurrent neural network (RNN) [20], and an advanced model based on bidirectional encoder representations from transformers (BERT) [50].

C. RNN-Based Model

An RNN-based model [20] proposed in 2016 was the first DNN-AES model. Fig. 1 shows the model architecture. This model calculates a score for a given essay, which is defined as a sequence of words, through the following multilayered neural networks.

1) *Lookup Table Layer:* This layer transforms each word in a given essay into a D-dimensional word-embedding representation, in which words with the same meaning have similar representations. Specifically, letting $\boldsymbol{w}_{jt}$ be a $|\mathcal{V}|$-dimensional one-hot representation of w_{jt}, and letting $\boldsymbol{A}$ be a $D \times |\mathcal{V}|$-dimensional embeddings matrix, the embedding representation corresponding to $w_{jt} \in e_j$ is calculable as the dot product $\boldsymbol{A} \cdot \boldsymbol{w}_{jt}$. Here, the one-hot representation of a word is a vector with a length equal to the size of the vocabulary and takes one for a single position corresponding to the index of that word and zero for other positions.

2) *Convolution Layer:* This layer extracts n-gram-level features using convolutional neural networks (CNNs) from the sequence of word-embedding vectors by transforming each word vector into another vector representation that reflects dependencies among n-adjacent words. Zero padding, an operation in which zeros are appended to the beginning of a sequence, is applied to the output sequence from this layer to preserve the word length. This is an optional layer, often omitted in current studies.
3) *Recurrent Layer:* This layer transforms each output vector from the convolution layer to another vector representation by using a long short-term memory (LSTM) network, a representative RNN, to consider the context of the target essay. A single-layer unidirectional LSTM is generally used, but bidirectional or multilayered LSTMs are also often used.
4) *Pooling Layer:* This layer transforms the output hidden vector sequence of the recurrent layer $\mathcal{H}_j = \{\boldsymbol{h}_{j1}, \boldsymbol{h}_{j2}, \ldots, \boldsymbol{h}_{jn_j}\}$ (where $\boldsymbol{h}_{jt}$ represents the output hidden vector of the recurrent layer for inputted word w_{jt}) into an aggregated fixed-length hidden vector. Mean-over-time (MoT) pooling, which calculates an average vector

$$\boldsymbol{M}_j = \frac{1}{n_j}\sum_{t=1}^{n_j} \boldsymbol{h}_{jt} \tag{3}$$

is generally used because it tends to provide stable accuracy. Other frequently used pooling methods include the last pool, which uses the last output of the recurrent layer $\boldsymbol{h}_{jn_j}$.
5) *Linear Layer With Sigmoid Activation:* This layer projects the output vector of the pooling layer to a scalar value in the range [0, 1] by utilizing the sigmoid function as

$$\sigma(\boldsymbol{W}\boldsymbol{M}_j + b) \tag{4}$$

where $\boldsymbol{W}$ is a weight matrix and b is a bias. Model training is conducted by normalizing gold-standard scores to $[0, 1]$, but the predicted scores are linearly rescaled to the original score range in the prediction phase.

D. BERT-Based Model

BERT, a pretrained language model released by the Google AI Language Team, has achieved state-of-the-art results in various NLP tasks [50]. It has been applied to AES [28], [51] and automated short-answer grading [52]–[54] since 2019 and provides good accuracy.

BERT is defined as a multilayer bidirectional transformer network [55]. Transformers are a neural network architecture designed to handle ordered sequences of data using an attention mechanism. Specifically, transformers consist of multiple layers (called *transformer blocks*), each containing a multihead self-attention and a positionwise fully connected feedforward network. This unique architecture enables transformers to consider relations among all pairs of elements in a sequence, thereby capturing its context more accurately than an RNN could. See [55] for details of transformers.

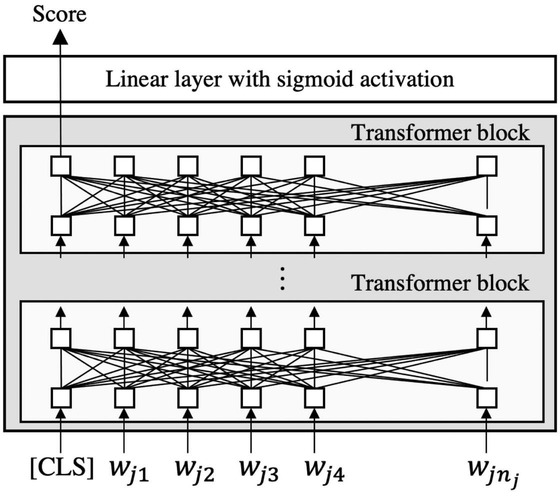


Fig. 2. BERT-based model architecture.

BERT is trained in *pretraining* and *fine-tuning* steps. Pretraining is conducted on huge amounts of unlabeled text data over two tasks, namely, *masked language modeling* and *next-sentence prediction*. Masked language modeling predicts the identities of words that have been masked out of the input text, while next-sequence prediction predicts whether two given sentences are adjacent.

Using BERT for a target NLP task, including AES, requires fine-tuning (retraining), which is conducted from a task-specific supervised dataset after initializing model parameters to pretrained values. When using BERT for AES, input essays require preprocessing, namely, adding a special token [CLS] to the beginning of each input. BERT output corresponding to this token is used as the aggregate hidden representation for a given essay [50]. We can, thus, score an essay by inputting its representation to a *linear layer with sigmoid activation*, as illustrated in Fig. 2.

E. Problems in Model Training

To use these supervised machine-learning-based AES models, they must be trained using a large dataset of essays that were graded by human raters. Note that BERT is a pretrained model, but it too requires fine-tuning using a graded essay dataset. For model training, the mean square error (MSE) between predicted and gold-standard scores is generally used as the loss function. Specifically, letting y_j be the gold-standard score for essay e_j and letting $\hat{y}_j$ be the predicted score, the MSE loss function is defined as

$$\frac{1}{J}\sum_{j=1}^{J}(y_j - \hat{y}_j)^2. \tag{5}$$

When only one rater is assigned to each essay, the gold-standard score y_j is the rating for essay e_j given by a single rater assigned from a set of raters $\mathcal{R}$. When multiple raters grade each essay, as is assumed in this study, the gold-standard score is usually determined by averaging multiple rater scores as

$$y_j = \frac{1}{R_j} \sum_{r \in \mathcal{R}, U_{jr} \neq -1} U_{jr} \tag{6}$$

where R_j represents the number of raters assigned to essay e_j. However, such simple scores depend strongly on rater characteristics, as discussed in Section I. The accuracy of an AES model drops when such biased data are used for model training because the trained model inherits bias effects [15], [36]–[38]. In educational and psychological measurement research, IRT models that can estimate essay scores while considering effects of rater characteristics have recently been proposed [7], [13], [39]–[42]. The main goal of this study is to train AES models using IRT-based unbiased scores. The next section introduces the IRT models.

IV. IRT Models

IRT [56], a test theory based on mathematical models, is widely used in educational testing. The primary purpose of a test theory is to estimate examinees' abilities from testing data, which generally consist of binary or polytomous scores that the examinees received on test items. IRT estimates examinees' abilities while considering the characteristics of the test items, including item difficulty and discrimination, by using probabilistic models called IRT models, whereas classical test theory typically estimates abilities based on a simple total or the average score.

A. Polytomous IRT Models

Traditional IRT models are applicable to two-way data consisting of scores that examinee receive on test items. For example, the generalized partial credit model (GPCM) [57], a representative polytomous IRT model, defines the probability that examinee j receives score k for test item i as

$$P_{ijk} = \frac{\exp \sum_{m=1}^{k} \left[D\alpha_i(\theta_j - \beta_i - d_{im}) \right]}{\sum_{l=1}^{K} \exp \sum_{m=1}^{l} \left[D\alpha_i(\theta_j - \beta_i - d_{im}) \right]} \tag{7}$$

where θ_j is the latent ability of examinee j, α_i is the discrimination parameter for item i, β_i is the difficulty parameter for item i, and d_{im} is the step difficulty parameter denoting difficulty of transition between scores $m-1$ and m in the item. $D = 1.7$ is the scaling constant used to minimize the difference between the normal and logistic distribution functions. Here, $d_{i1} = 0$, and $\sum_{m=2}^{K} d_{im} = 0$ is given for model identification.

The key feature of IRT models, including GPCM, is that they represent the probability of each observed score as a function of latent examinee ability and item characteristics. The parameters of ability and item characteristics can be estimated from a collection of such observed scores. IRT models generally provide more accurate estimates of ability compared with classical test theory because they can estimate examinee ability while considering the effects of item characteristics.

However, traditional IRT models ignore rater factors; therefore, they are not applicable to rating data from multiple raters, as assumed in this study. Extension models that incorporate parameters representing rater characteristics have been proposed to resolve this limitation [13], [39]–[43].

B. IRT Models With Rater Parameters

This study introduces one of the newest models, namely, the generalized many-facet Rasch model (GMFRM) [42], [43]. The GMFRM defines the probability that rater r assigns score k to examinee j's essay for a test item (e.g., an essay task or a prompt) i as

$$P_{ijrk} = \frac{\exp \sum_{m=1}^{k} \left[D\alpha_r\alpha_i(\theta_j - \beta_r - \beta_i - d_{rm}) \right]}{\sum_{l=1}^{K} \exp \sum_{m=1}^{l} \left[D\alpha_r\alpha_i(\theta_j - \beta_r - \beta_i - d_{rm}) \right]} \tag{8}$$

where α_r is the consistency of rater r, β_r is the severity of rater r, and d_{rm} represents the strictness of rater r for category m. For model identification, $\sum_{i=1}^{I} \log \alpha_i = 0$, $\sum_{i=1}^{I} \beta_i = 0$, $d_{r1} = 0$, and $\sum_{m=2}^{K} d_{rm} = 0$ are assumed. The GMFRM is expected to be robust for a large variety of raters [42], [43] because it can consider various types of rater bias effect, as described in the next section.

This study, therefore, assumes the application of a GMFRM to rating data $\boldsymbol{U}$ in training data. Note that, in general, AES models are independently trained for each essay task. Therefore, rating data $\boldsymbol{U}$ are defined as two-way data in Section II. When the number of tasks is fixed to one in the GMFRM, the above mentioned model identification constraints make α_i and β_i ignorable; therefore, (8) becomes

$$P_{jrk} = \frac{\exp \sum_{m=1}^{k} \left[D\alpha_r(\theta_j - \beta_r - d_{rm}) \right]}{\sum_{l=1}^{K} \exp \sum_{m=1}^{l} \left[D\alpha_r(\theta_j - \beta_r - d_{rm}) \right]}. \tag{9}$$

This equation is consistent with the conventional GPCM, regarding the item parameters as the rater parameters. Note that θ_j in (9) represents not only the ability of examinee j, but also the latent unbiased scores for essay e_j, because only one essay is associated with each examinee.

The unbiased essay scores θ_j in the model can be estimated from observed essay rating data $\boldsymbol{U}$ while considering rater bias effects in a manner similar to that of the traditional GPCM, which can estimate examinee abilities while considering the effects of item characteristics. IRT models with rater parameters, including GMFRM, have been widely used for various performance tests, including essay writing tests and speaking tests, not only to realize an accurate ability or score estimation but also to analyze effects of various bias factors, such as rater bias (see, e.g., [8]–[13], [35], [41]–[43], and [58]).

C. Rater Biases

The GMFRM given as (9) can consider rater biases induced by differences in the following three common rater characteristics [1], [9]–[11], [31], [35], [37], [59]–[61].

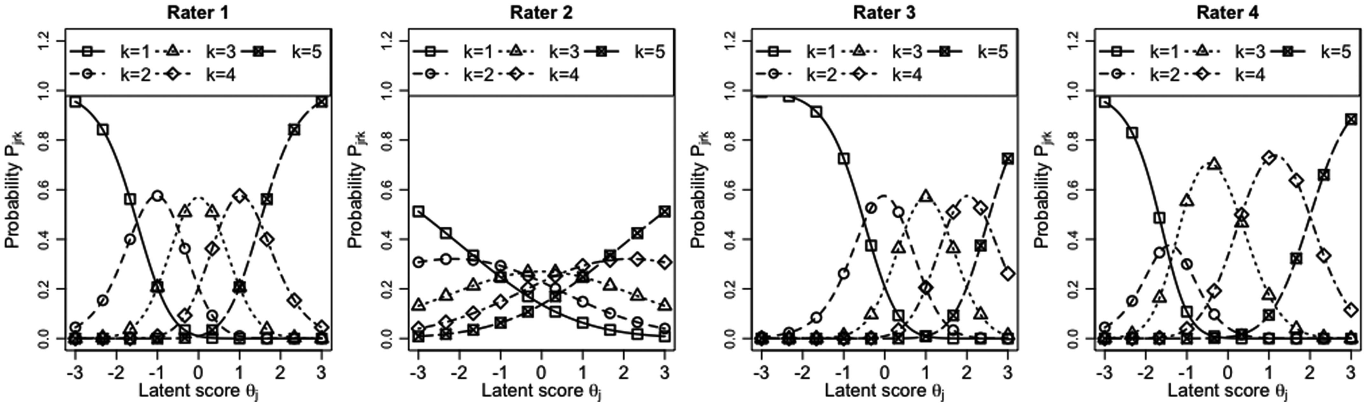


Fig. 3. IRCs of four raters with the different parameters presented in Table I.

TABLE I
PARAMETERS FOR FOUR RATERS WITH DIFFERENT CHARACTERISTICS

	α_r	β_r	d_{r2}	d_{r3}	d_{r4}	d_{r5}
Rater 1	1.2	0.0	-1.5	-0.5	0.5	1.5
Rater 2	0.2	0.0	-1.5	-0.5	0.5	1.5
Rater 3	1.2	1.0	-1.5	-0.5	0.5	1.5
Rater 4	1.2	0.0	-1.5	-1.3	0.3	2.0

1) *Consistency:* The extent to which the rater assigns similar ratings to essays of similar quality [9].
2) *Severity:* The tendency to give consistently lower ratings than that are justified by the essays [9].
3) *Range Restriction:* The tendency to overuse a few rating categories [9], [10], [31]. The central tendency (a tendency to overuse central categories) and the extreme response tendency (a tendency to prefer endpoints of the response scale) are special cases of range restriction.

To show how these characteristics are represented in the GMFRM, Fig. 3 depicts item response curves (IRCs) of the GMFRM, which are drawn by plotting the probability P_{jrk} in (9) for four raters with the different parameters presented in Table I. In Fig. 3, the horizontal axis shows the latent score θ_j and the vertical axis shows the probability P_{jrk}. These IRCs show that essays with lower (higher) θ_j tend to obtain lower (higher) scores.

The GMFRM represents rater consistency as α_r, with lower values indicating smaller differences in response probabilities between rating categories. This can be confirmed in Fig. 3 comparing raters 1 and 2, who have different consistency levels. Fig. 3 suggests that scores given by a rater with a lower consistency parameter will be strongly biased because they tend to assign different ratings to essays with similar qualities.

In the GMFRM, rater severity is represented as β_r. The IRC shifts to the right as this parameter value increases, indicating that raters with high β_r tend to consistently assign low scores. In Fig. 3, the IRC for rater 3 with a high β_r value shifts to the right overall. This indicates that scores given by raters with high or low severity are biased.

The GMFRM represents the range restriction characteristic as d_{rm}. When $d_{r(m+1)}$ and d_{rm} are closer, the probability of responding with category m decreases overall. Conversely, as the difference $d_{r(m+1)} - d_{rm}$ increases, the response probability for category m increases. In Fig. 3, rater 4 has a smaller $d_{r3} - d_{r2}$ value and relatively larger $d_{r4} - d_{r3}$ and $d_{r5} - d_{r4}$ values. Thus, in the IRC, response probabilities for category 2 decrease, whereas those for categories 3 and 4 increase, representing a range restriction characteristic with overuse of categories 3 and 4 while avoiding category 2. A strong range restriction characteristic causes biased rating data because it means the rater's score distribution differs extremely from the other raters.

The GMFRM can estimate the latent essay scores θ_j while considering these rater bias effects. Note that the GMFRM cannot directly represent several rater biases, including differential rater functioning and rater drift [62], [63], although it can capture various rater characteristics, as described previously.

D. IRT Parameter Estimation

The model parameters in (9) can be estimated from rating data U. As the parameter estimation method for traditional IRT models, marginal maximum likelihood estimation using an expectation maximization algorithm has been widely used [64]. However, for complex models such as the GMFRM, expected *a posteriori* (EAP) estimation, a type of Bayesian estimation, is known to provide more robust estimations [41], [65]. The GMFRM, therefore, uses EAP estimation based on the No-U-Turn (NUT) sampler algorithm [66]. The NUT sampler is a Markov chain Monte Carlo (MCMC) algorithm that improves greater efficiency as compared with the Metropolis-Hastings-within-Gibbs sampler [40], a conventional MCMC algorithm for IRT models [67]. The estimation program for the GMFRM, which is implemented in RStan [68], [69], was previously published in [43]. Following the original GMFRM paper, we assume the standard normal distribution $N(0.0, 1.0)$ as a prior distribution for θ_j, $\log \alpha_r$, β_r, and d_{rk}. In addition, we calculate the EAP estimates using parameter samples obtained from 2000–4000 periods.

V. PROPOSED METHOD

As described above, the main idea of this study is to train AES models using IRT-based scores $\boldsymbol{\theta} = \{\theta_j \mid j \in \mathcal{J}\}$ to deal with rater bias in training data. In the proposed method, we can use any regression-based AES model, including those introduced in Section III.

A. Model Training

Training of the proposed method occurs in the following two steps.

1) Estimate the IRT scores $\boldsymbol{\theta}$ from the rating data U. This study uses an MCMC algorithm for this estimation, as described in Section IV-D.
2) Train AES models using the IRT scores $\boldsymbol{\theta}$ as the gold-standard scores. Specifically, the MSE loss function for training is defined as

$$\frac{1}{J}\sum_{j=1}^{J}(\theta_j - \hat{\theta}_j)^2 \tag{10}$$

where $\hat{\theta}_j$ represents the AES's predicted score for essay e_j. Note that the gold-standard scores must be rescaled to the range $[0, 1]$ for training when a DNN-based AES model is used because it uses the sigmoid activation in the output layer. In IRT, 99.7% of θ_j fall within the range $[-3, 3]$ on the logit scale, because a standard normal distribution is generally assumed. We, therefore, apply a linear transformation from the logit range $[-3, 3]$ to $[0, 1]$ after adjusting scores lower than -3 to -3 and those higher than 3 to 3.

Because IRT-based scores $\boldsymbol{\theta}$ are estimated while removing rater bias effects, a trained AES model based on this method will not reflect bias effects.

B. Score Prediction

In the prediction phase, the score for new essay $e_{j'}$ is calculated in the following two steps.

1) Predict the IRT score $\theta_{j'}$ from a trained AES model. Then, linearly rescale it to the logit range $[-3, 3]$ when a DNN-based AES model is used.
2) Calculate the expected score $\hat{U}_{j'}$, which corresponds to an unbiased original-scaled score of e'_j [32], given $\theta_{j'}$ and rater parameters as

$$\hat{U}'_j = \frac{1}{R}\sum_{r=1}^{R}\sum_{k=1}^{K} k \cdot P_{j'rk}. \tag{11}$$

The expected score $\hat{U}'_j$ can be used as a predicted essay score of the proposed method.

The proposed method can also predict each rating, although this prediction is not the main objective of this study. Specifically, rating of rater r for essay $e_{j'}$ can be predicted based on the IRT model given $\theta_{j'}$ and rater parameters as

$$\hat{U}_{j'r} = \sum_{k=1}^{K} k \cdot P_{j'rk}. \tag{12}$$

VI. Experiments

This section describes evaluations of the effectiveness of the proposed method through actual-data experiments.

A. Actual Data

Our experiments use the ASAP dataset, which is widely used as benchmark data in AES studies. This dataset consists of essays on eight topics, written by students from grades 7 to 10. There are 12 978 essays, averaging 1622 essays per topic. However, this dataset cannot be directly used to evaluate the proposed method, because despite its essays having been graded by multiple raters, it contains no rater identifiers.

We, therefore, employed other raters and asked them to grade essays in the ASAP dataset. We used essay data for the fifth ASAP topic, because the number of essays in that topic is relatively large ($n = 1805$). We recruited 38 native English speakers as raters through Amazon Mechanical Turk and assigned four raters to each essay to decrease rater workloads. The rater assignment was conducted based on a systematic links design [70]–[72] to achieve IRT-scale linking. As a result, each rater graded around 195 essays. We asked the raters to grade following the same assessment rubric as that used for creating the original ASAP dataset. The rubric is a holistic rubric (a single-criterion rubric used to assess overall essay quality) with five rating categories. The average Pearson's correlation between the rating scores collected in this experiment and the original ASAP scores was 0.675.

B. Analysis of Rater Biases

To confirm what differences in rater characteristics exist, Table II shows descriptive statistics of rating data for each rater and rater parameter estimates in the IRT model defined by (9). Table II shows that these descriptive statistics and the IRT-based rater parameters vary across raters, which reflects that the raters have different rating behaviors. As examples, Fig. 4 depicts the IRCs of raters 3, 16, 31, and 34. The horizontal axis shows the latent score θ_j, and the vertical axis shows the response probability of the rater for each category. According to Table II and Fig. 4, the characteristics of each rater can be interpreted as follows.

1) Rater 3 has average levels of consistency and severity. Furthermore, the appearance frequency distribution of the rating categories is similar to the averaged one shown in the last row of Table II. This rater can, thus, be considered as having a standard rating characteristic.
2) As the score appearance frequency and the IRC show, rater 16 tends to prefer extreme scores (1 and 5), as compared with the other raters. This is a typical example of the extreme response tendency, where a rater tends to overuse the extreme rating categories while avoiding the middle categories.
3) Rater 31 shows a high average score value and a low rater severity value. This rater tends to overuse high scores (4 and 5), as shown in the IRC. This suggests that this rater is extremely lenient overall.
4) Rater 34 has a low consistency value α_r. In this rater's IRC, the differences in response probabilities among categories are small as compared with raters with high consistency levels, such as Rater 3. This rater, thus, has a stronger tendency to assign different ratings to essays

TABLE II
DESCRIPTIVE STATISTICS AND IRT PARAMETERS FOR EACH RATER, CALCULATED FROM ACTUAL DATA

Raters	Descriptive statistics							IRT-based rater parameters					
	Avg.	*SD*	Appearance frequency of each rating category					α_r	β_r	d_{r2}	d_{r3}	d_{r4}	d_{r5}
			1	2	3	4	5						
1	3.32	1.13	8	45	53	55	34	1.79	-0.28	-1.57	-0.27	0.49	1.36
2	3.29	1.15	17	29	57	64	28	0.81	-0.15	-1.17	-0.72	0.28	1.60
3	3.53	1.11	10	20	68	51	46	1.54	-0.48	-1.05	-0.71	0.55	1.21
4	3.43	1.15	11	30	59	54	41	1.03	-0.55	-1.42	-0.53	0.59	1.36
5	3.57	0.85	7	6	70	93	19	0.49	-0.70	-0.62	-2.54	0.23	2.93
6	4.31	0.94	3	11	14	62	105	1.84	-1.26	-1.22	-0.24	0.11	1.34
7	4.00	1.09	4	22	26	62	81	1.14	-1.22	-1.51	-0.09	0.26	1.35
8	4.37	0.97	4	8	20	43	120	1.18	-1.61	-0.88	-0.44	0.31	1.01
9	3.27	1.01	6	39	69	58	23	2.08	-0.39	-1.83	-0.43	0.65	1.61
10	4.49	0.75	0	4	18	51	122	1.50	-1.89	-1.28	-0.57	0.42	1.42
11	3.40	1.15	11	31	64	48	41	1.39	-0.46	-1.37	-0.50	0.61	1.26
12	4.22	0.94	3	7	31	58	97	1.05	-1.32	-0.95	-0.75	0.42	1.27
13	3.19	1.00	6	42	80	44	23	1.17	-0.26	-1.94	-0.54	0.88	1.60
14	3.34	1.07	12	32	49	81	21	1.81	-0.42	-1.39	-0.53	0.19	1.73
15	3.87	1.15	6	26	31	58	75	1.22	-0.83	-1.26	-0.16	0.26	1.16
16	3.94	1.46	24	16	21	20	114	0.60	-0.92	0.08	-0.02	0.51	-0.58
17	3.56	0.97	4	23	62	74	33	1.22	-0.76	-1.75	-0.60	0.56	1.79
18	3.20	1.17	17	39	56	55	28	1.87	-0.24	-1.56	-0.43	0.51	1.48
19	3.03	1.00	11	47	77	46	14	1.95	-0.04	-1.85	-0.57	0.71	1.70
20	3.74	1.08	3	26	50	56	60	2.03	-0.75	-1.65	-0.32	0.58	1.38
21	3.78	1.16	6	27	40	53	69	1.88	-0.68	-1.32	-0.24	0.44	1.12
22	3.65	1.20	10	30	35	63	57	2.95	-0.49	-1.16	-0.25	0.26	1.14
23	3.87	1.03	5	15	43	69	63	1.81	-1.01	-1.42	-0.58	0.47	1.54
24	3.55	0.87	1	22	65	84	24	1.01	-0.76	-2.02	-0.71	0.51	2.22
25	4.29	0.95	2	9	28	47	109	1.17	-1.48	-1.00	-0.48	0.42	1.06
26	3.54	1.21	14	28	40	64	49	2.14	-0.56	-1.21	-0.35	0.32	1.24
27	3.15	1.21	17	47	54	44	33	0.99	-0.16	-1.47	-0.29	0.55	1.20
28	3.74	1.02	5	14	61	62	53	1.41	-0.85	-1.23	-0.81	0.57	1.47
29	3.28	1.23	21	31	50	59	34	0.81	-0.25	-1.03	-0.56	0.26	1.34
30	3.35	0.92	4	29	76	67	19	1.02	-0.52	-2.11	-0.75	0.74	2.11
31	4.45	0.71	0	4	13	69	109	1.14	-1.76	-1.22	-0.57	0.11	1.68
32	3.45	1.01	3	33	66	60	33	1.45	-0.53	-1.96	-0.39	0.70	1.64
33	3.58	1.11	8	27	50	64	46	1.19	-0.55	-1.35	-0.45	0.41	1.39
34	3.26	1.46	1	35	31	26	51	0.56	-0.27	-0.52	0.02	-0.30	0.81
35	3.73	1.19	5	32	44	43	71	1.24	-0.89	-1.67	-0.16	0.69	1.14
36	4.01	0.97	4	12	31	80	68	1.31	-0.93	-1.33	-0.44	0.28	1.50
37	3.08	1.22	18	50	60	33	34	0.85	-0.13	-1.46	-0.30	0.85	0.91
38	4.29	0.75	0	2	29	75	89	1.17	-1.61	-1.27	-1.04	0.51	1.80
Average	3.66	1.06	7.66	25.00	47.13	57.76	56.21	1.36	-0.74	-1.34	-0.51	0.45	1.40

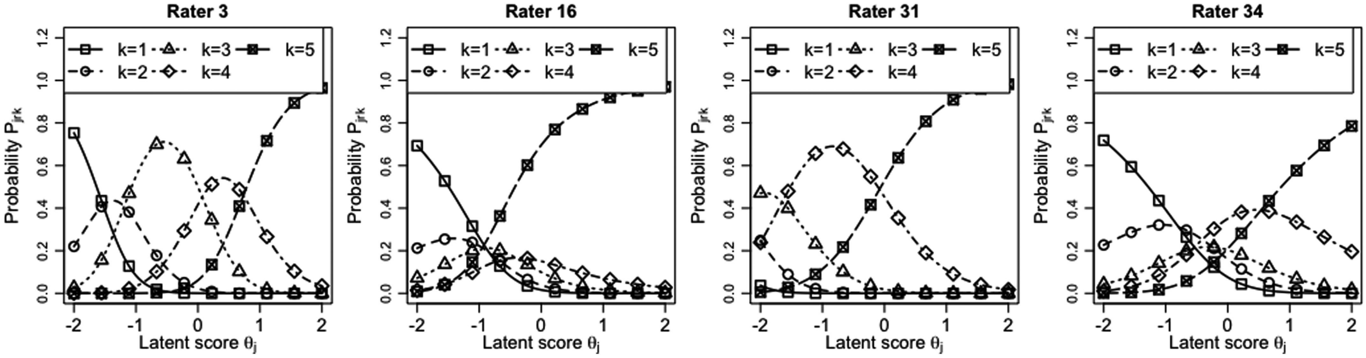


Fig. 4. IRCs of four representative raters found in actual-data experiments.

with similar quality. Such raters generally lower the assessment accuracy because their ratings do not necessarily reflect the true essay quality.

As these examples show, we can confirm that the rating characteristics differed among the raters.

To quantitatively examine whether consideration of all three rater characteristics assumed in the GMFRM is effective, we conducted a model comparison experiment. In that experiment, we compared information criteria among the GMFRM and their restricted versions. We used the following three models as restricted versions.

1) *Consistency-Fixed Model*: It is a model in which α_r is restricted to one for all raters $r \in \mathcal{R}$, meaning that all raters share the same consistency level.

TABLE III
INFORMATION CRITERIA FOR THE GMFRM AND COMPARATIVE MODELS

	WAIC	WBIC
GMFRM	**13 696.82**	10 983.61
Consistency fixed model	14 404.27	11 463.23
Severity fixed model	15 098.69	11 909.43
Threshold fixed model	14 138.21	**10 780.76**
MFRM	14 863.44	11 313.59

2) *Severity-Fixed Model*: It is a model in which β_r is restricted to zero for all raters $r \in \mathcal{R}$, meaning that all raters share the same severity level.
3) *Threshold-Fixed Model*: It is a model in which d_{rm} is changed to d_m for all raters $r \in \mathcal{R}$, meaning that no difference in range restriction characteristics exists among raters.

We also compared the GMFRM with a many-facet Rasch model (MFRM) [73], which is the most popular IRT model that incorporates rater parameters. Although the MFRM has several forms, this experiment used the simplest one, which is equivalent to a GMFRM in which α_r is restricted to one and d_{rm} is changed to d_m for all raters. Note that the *consistency-fixed model* defined above is also equivalent to another form of MFRM.

As information criteria, the Akaike information criterion (AIC) [74], the Bayesian information criterion (BIC) [75], the widely applicable information criterion (WAIC) [76], and the widely applicable Bayesian information criterion (WBIC) [77] are often used. The AIC and the BIC are applicable when maximum likelihood estimation is used to estimate model parameters, whereas the WAIC and the WBIC can be used with Bayesian estimation using MCMC or variational inference methods. Because this study uses Bayesian estimation based on MCMC, as described in Section IV-D, this experiment uses the WAIC and the WBIC. The model minimizing these criteria values is regarded as the optimal model.

Table III shows the results, with bold text indicating minimum scores and underlined text representing second smallest scores for each criterion. According to the results, the criteria value increases when one of the three rater characteristic parameters is removed from the GMFRM, except for the WBIC for the *threshold-fixed model*. The results suggest that the three characteristics vary among the raters although the difference in the range restriction might be relatively small. Furthermore, the GMFRM outperformed the MFRM on both criteria, suggesting that the GMFRM is more suitable for the data.

The analysis described in this subsection demonstrates that consideration of rater bias is required to realize a robust AES, because the assessment characteristics differed among raters.

C. Model–Data Fit and IRT Score Estimates

The model comparison experiment described above demonstrates that the GMFRM is more suitable compared with its restricted models and the MFRM. However, we should also check the goodness of the model–data fit for the GMFRM itself. To examine the model–data fit, we used a posterior predictive p-value (PPP-value) [78], which is commonly used to evaluate model–data fit in Bayesian frameworks [79], [80]. Specifically, we calculated a PPP-value for the GMFRM by using an averaged standardized residual (a traditional metric of IRT model fitness under a non-Bayesian framework) as a discrepancy function, in a manner similar to that in [80] and [81]. The PPP-value takes around 0.5 for a well-fitted model but takes extreme low or high values, such as those less than 0.05 or higher than 0.95, for a poorly fitted model. As a result, the PPP-value of the GMFRM was 0.57, which is near 0.5, suggesting that the model is well fitted to the data.

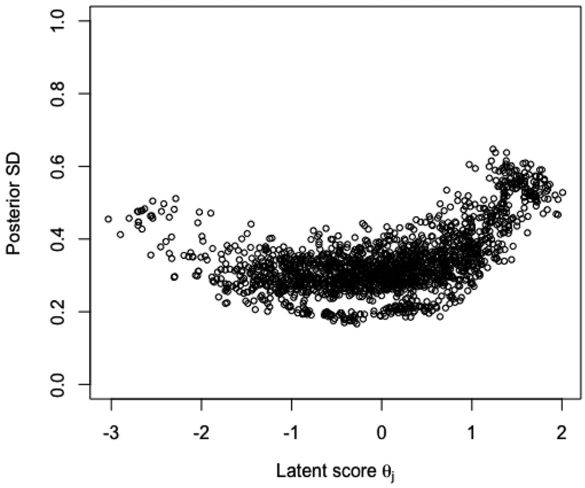


Fig. 5. IRT score estimates and corresponding posterior SD.

Furthermore, to check how accurately the GMFRM estimated the latent score θ_j, we calculated the posterior standard deviation (SD), which is the Bayesian analog of the standard error. Fig. 5 shows the results. In the figure, the horizontal axis shows the point estimates of θ_j, the vertical axis shows corresponding posterior SD values, and each plot indicates an essay. Because the standard normal distribution is assumed for the IRT scores as described earlier, the estimated θ_j values were distributed around zero in the figure. The averaged posterior SD value was 0.338, which should be acceptable, especially in educational contexts or low- and medium-stake tests, because it corresponds to only about 5% of the logit range $[-3, 3]$ where 99.7% of θ_j falls statistically.

As described in this section, in practice, it is preferable to check the model–data fit and the estimation errors for IRT scores when an IRT model is applied to rating data at the first step of the training process in the proposed method.

D. Evaluating Robustness of Score Prediction

This section evaluates whether the proposed method can provide more robust scores than can the conventional AES models, even when the rater assignment for each essay in the training data changes. The experimental procedures, which were inspired by those used in previous studies examining IRT scoring robustness [32], [33], [41], [44] and outlined in Fig. 6, are as follows.

1) We estimated rater parameters in the IRT model by an MCMC algorithm using all rating data.

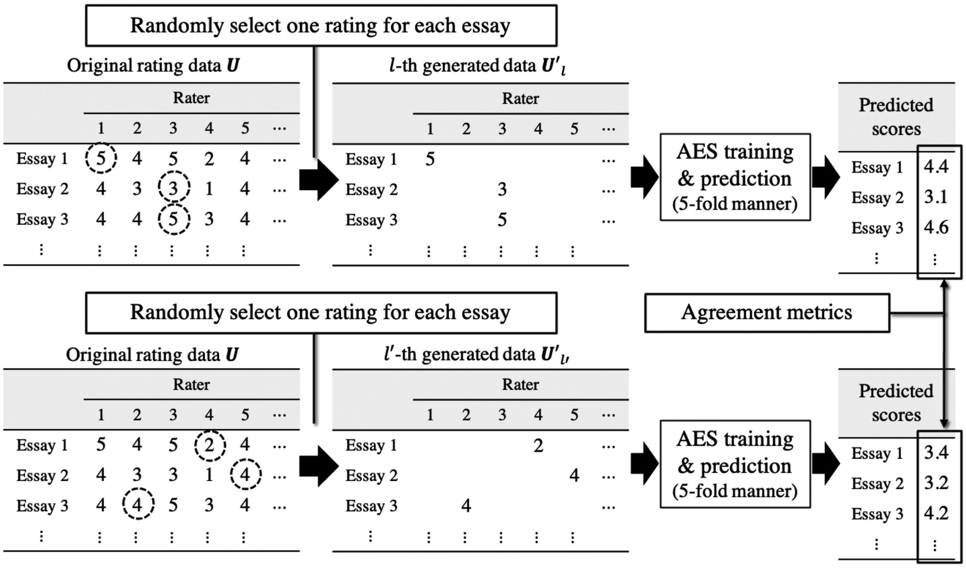


Fig. 6. Outline of the robustness evaluation experiment.

2) We created a dataset consisting of essay rating pairs by randomly selecting one rating for each essay from the ratings assigned by multiple raters. We repeated this data generation ten times. Hereafter, the lth generated dataset is represented as $\boldsymbol{U}'_l$ $(l \in \{1, \ldots, 10\})$.
3) From each dataset $\boldsymbol{U}'_l$, we estimated IRT scores (referred to as $\boldsymbol{\theta}_l$) given the rater parameters obtained in Procedure 1) and, then, created a dataset $\boldsymbol{U}''_l$ consisting of essay-IRT-based score pairs.
4) Using each dataset $\boldsymbol{U}''_l$, we applied a fivefold method to train AES models and to obtain predicted scores $\hat{\boldsymbol{\theta}}_l$ for all essays. Specifically, in each step of the fivefold method, we trained an AES model using 80% of essays and predicted scores for the remaining 20% of essays. We repeated the step five times to obtain predicted scores for all essays.
5) We calculated metrics for agreement between the expected scores calculated by (11) given $\hat{\boldsymbol{\theta}}_l$ and those calculated given $\hat{\boldsymbol{\theta}}_{l'}$ for all unique $l, l' \in \{1, \ldots, 10\}$ pairs (${}_{10}C_2 = 45$ pairs in total). As agreement metrics, we used mean absolute error (MAE), root-mean-square error (RMSE), Pearson correlation coefficient, accuracy, Cohen's kappa, and linear weighted kappa. Note that we calculated the accuracy, kappa, and weighted kappa by rounding the expected scores.
6) We calculated average metric values obtained from the 45 pairs.

Better agreements—namely those with high accuracy, kappa, weighted kappa, and correlation and low MAE and RMSE values—indicate that the method outputs stable scores even when the rater assignment in the training dataset is changed, meaning that it is robust for different raters.

We conducted a similar experiment using conventional AES models without the IRT model. Specifically, using each dataset $\boldsymbol{U}'_l$ created in Procedure 2), we predicted essay scores from an AES model through the fivefold method as in Procedure 4). We then calculated the six agreement metrics among the predicted scores obtained from different datasets $\boldsymbol{U}'_l$ and $\boldsymbol{U}'_{l'}$ for all unique $l, l' \in \{1, \ldots, 10\}$ pairs and averaged the metric values.

TABLE IV
VARIANTS OF RNN-BASED AES MODELS

	Convolution Layer	Recurrent Layer	Pooling Layer
LSTM (Last)	Not used	LSTM	Last pooling
LSTM (MoT)	Not used	LSTM	Mean over time
CNN+LSTM (MoT)	Used	LSTM	Mean over time
2L-LSTM (MoT)	Not used	2-layer LSTM	Mean over time
Bidirectional LSTM	Not used	BiLSTM	Last pooling

We also conducted Student's t-test for the averaged agreement metrics between the proposed method and the conventional method.

These experiments were conducted with several AES models. We used EASE [47] as a feature engineering-based model, because as mentioned in Section III-A, it is known to provide high performance. As DNN-based automatic feature extraction models, we examined several variants of RNN-based models introduced in Section III-C and the BERT-based model introduced in Section III-D. Table IV summarizes settings for the RNN-based model variants. These models were implemented in Python with the Keras library. The hyperparameters and dropout settings were determined following [20], [50], and [55]. Specifically, for RNN-based models, we set LSTM hidden-variable dimensions to 300, the mini batch size to 32, and the maximum epochs to 50. Word-embedding dimensions were set to 50, and the number of vocabulary words was set to 4000. We used dropout regularization to avoid overfitting, with dropout probabilities for lookup-table-layer output and pooling-layer output set to 0.5. We set the recurrent dropout probability to 0.1. We used the Adam optimization algorithm [82] to minimize the MSE loss function over the training data. For the BERT model, we used a *base*-sized pretrained model and fine-tuned given the mini batch size of 32 and maximum epochs of three.

TABLE V
RESULTS OF ROBUSTNESS EVALUATION

	MAE			RMSE			Correlation		
	Prop.	Conv.	*p*	Prop.	Conv.	*p*	Prop.	Conv.	*p*
DNN-based approach									
LSTM (Last)	**0.253**	0.462	< .01	**0.329**	0.607	< .01	**.788**	.735	< .01
LSTM (MoT)	**0.162**	0.286	< .01	**0.213**	0.382	< .01	**.930**	.908	< .01
CNN+LSTM (MoT)	**0.271**	0.517	< .01	**0.354**	0.689	< .01	**.829**	.762	< .01
2L-LSTM (MoT)	**0.157**	0.341	< .01	**0.208**	0.455	< .01	**.934**	.878	< .01
Bidirectional LSTM	**0.278**	0.518	< .01	**0.355**	0.677	< .01	**.773**	.689	< .01
BERT	**0.121**	0.233	< .01	**0.159**	0.311	< .01	**.960**	.935	< .01
Feature-engineering approach									
EASE	**0.062**	0.094	< .01	**0.081**	0.307	< .01	**.985**	.917	< .01
	Accuracy			Kappa			Weighted kappa		
	Prop.	Conv.	*p*	Prop.	Conv.	*p*	Prop.	Conv.	*p*
DNN-based approach									
LSTM (Last)	**.812**	.531	< .01	**.569**	.322	< .01	**.589**	.484	< .01
LSTM (MoT)	**.870**	.690	< .01	**.719**	.561	< .01	**.744**	.697	< .01
CNN+LSTM (MoT)	**.766**	.506	< .01	**.526**	.326	< .01	**.589**	.522	< .01
2L-LSTM (MoT)	**.880**	.627	< .01	**.741**	.471	< .01	**.765**	.633	< .01
Bidirectional LSTM	**.775**	.497	< .01	**.507**	.272	< .01	**.544**	.444	< .01
BERT	**.905**	.741	< .01	**.790**	.629	< .01	**.808**	.743	< .01
Feature-engineering approach									
EASE	**.940**	.906	< .01	**.886**	.851	< .01	**.894**	.881	< .01

High accuracy, kappa, weighted kappa, and correlation and low MAE and RMSE indicate high performance.

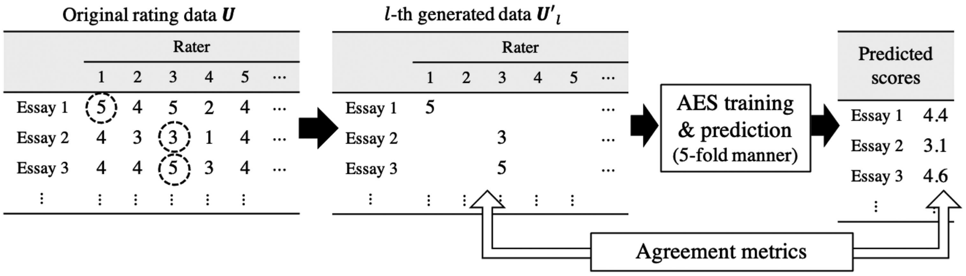


Fig. 7. Outline of rating prediction accuracy evaluation.

Table V shows the results. In Table V, bold text indicates better agreement between the proposed method and the conventional method. The table shows that the proposed method significantly improves agreement metrics compared with the conventional models in all cases. These results indicate that the proposed method provides stable scores when the rater assignment for each essay in the training data is changed, thus demonstrating that it is highly robust against rater bias.

Table V also shows that EASE provided the best agreement among the AES models in almost all cases. However, this does not necessarily mean that the proposed method with the EASE model provides the best performance. Section VI-F presents a detailed discussion of this point.

E. Evaluating the Accuracy of Rating Predictions

As described in Section V-B, the proposed method can predict each rating U_{jr} while considering the rater characteristics. If the proposed method works appropriately, the accuracy of the rating prediction is also expected to be improved. This subsection, therefore, evaluates this accuracy. The experimental procedures for the proposed method, shown in Fig. 7, are as follows.

1) We applied Procedures 1)–3) in the previous experiment.
2) In Procedure 4) of the previous experiment, we predicted each rating using (12) instead of calculating the expected score using (11).
3) We calculated the six agreement metrics between the predicted ratings and the gold-standard ratings in U'_l.
4) We repeated this ten times and calculated the averaged agreement values.

Similarly, the experimental procedures for the conventional method are as follows.

1) We conducted Procedures 1) and 2) in the previous experiment.
2) Using a dataset in $\{U'_1, \ldots, U'_{10}\}$, we predicted essay scores using the conventional AES method without consideration of rater effects through the fivefold method as in Procedure 4) in the previous experiment.
3) The remaining procedures were the same as Procedures 3) and 4) for the proposed method described above.

To evaluate differences in the averaged agreement metrics between the proposed and conventional methods, we conducted Student's *t*-test for each metric.

Table VI shows the results, with bold text indicating higher performance between the proposed and conventional methods. Comparing the DNN-AES models, Table VI indicates that the use of MoT pooling achieved higher performance than the use

TABLE VI
RESULTS OF RATING PREDICTION ACCURACY EVALUATION

	MAE			RMSE			Correlation		
	Prop.	Conv.	p	Prop.	Conv.	p	Prop.	Conv.	p
DNN-based approach									
LSTM (Last)	**0.717**	0.804	< .01	**0.900**	1.011	< .01	**.622**	.521	< .01
LSTM (MoT)	**0.614**	0.676	< .01	**0.769**	0.853	< .01	**.756**	.677	< .01
CNN+LSTM (MoT)	**0.645**	0.757	< .01	**0.811**	0.963	< .01	**.712**	.603	< .01
2L-LSTM (MoT)	**0.612**	0.691	< .01	**0.766**	0.868	< .01	**.757**	.667	< .01
Bidirectional LSTM	**0.739**	0.853	< .01	**0.926**	1.067	< .01	**.594**	.467	< .01
BERT	**0.597**	0.656	< .01	**0.750**	0.821	< .01	**.773**	.702	< .01
Feature-engineering approach									
EASE	**0.633**	0.638	.18	**0.792**	0.894	< .01	**.741**	.627	< .01
	Accuracy			Kappa			Weighted kappa		
	Prop.	Conv.	p	Prop.	Conv.	p	Prop.	Conv.	p
DNN-based approach									
LSTM (Last)	**.417**	.386	< .01	**.200**	.164	< .01	**.346**	.320	.09
LSTM (MoT)	**.479**	.457	< .01	**.286**	.265	< .01	**.455**	.454	.94
CNN+LSTM (MoT)	**.463**	.411	< .01	**.266**	.209	< .01	**.435**	.401	< .01
2L-LSTM (MoT)	**.479**	.449	< .01	**.285**	.255	< .01	**.456**	.443	.14
Bidirectional LSTM	**.404**	.355	< .01	**.181**	.122	< .01	**.327**	.274	< .01
BERT	**.498**	.473	< .01	**.311**	.285	< .01	**.477**	.474	.49
Feature-engineering approach									
EASE	**.452**	.441	.02	**.248**	.235	.04	**.407**	.405	.70

High accuracy, kappa, weighted kappa, and correlation and low MAE and RMSE indicate high performance.

of last pooling. The table also shows that the CNN did not effectively improve accuracy. These tendencies are consistent with a previous study [20]. In addition, the BERT provided the highest accuracy, which is also consistent with current NLP studies [50]–[54]. EASE too provided good performance but was inferior to some DNN-AES models.

The results also show that the proposed method provided sufficiently higher performance than the conventional method in almost all cases. The IRT-based scores reflect essay qualities more accurately than do the observed raw ratings, because rater biases are removed. Use of the IRT-based scores, thus, makes it easier for the AES models to capture relations between textual information and scores. In addition, the proposed method can predict each rating while considering rater characteristics, whereas the conventional method ignores them. These are reasons why the proposed method improved rating prediction accuracies.

As an alternative approach to predict each rating as in this experiment, a clustering-based method [14] has been proposed. The idea of this method is to predict each rating after estimating the raters who are assigned to each essay when we lack information about which rater graded which essay within the training data. When we know an actual rater assignment, our method is expected to provide higher accuracy because it can use this information directly. Although, unlike our method, this method cannot predict scores that mitigate rater bias effects, a comparison of rating prediction accuracy will be performed in a future work.

TABLE VII
STATISTICS OF OUTPUT SCORES

	Prop.			Conv.		
	Avg.	*SD*	$\bar{\sigma}$	*Avg.*	*SD*	$\bar{\sigma}$
LSTM (Last)	3.71	0.50	0.21	3.75	0.83	0.38
LSTM (MoT)	3.70	0.55	0.13	3.70	0.88	0.24
CNN+LSTM (MoT)	3.70	0.59	0.22	3.66	0.98	0.43
2L-LSTM (MoT)	3.70	0.56	0.13	3.68	0.89	0.28
Bidirectional LSTM	3.73	0.52	0.23	3.75	0.86	0.43
BERT	3.72	0.55	0.10	3.72	0.86	0.19
EASE	3.68	0.47	0.05	3.64	0.75	0.10

F. Discussion

The experimental results presented in Sections VI-D and VI-E show that a DNN-AES model with higher robustness tends to provide higher rating prediction accuracies. However, the EASE model shows a different tendency. Specifically, EASE presented the best agreement values in the first experiment, but not in the second experiment. To discuss this point in more detail, we calculated the average and SD of the output scores obtained from each AES model. The *Avg.* and SD columns in Table VII show the results.

Table VII shows that EASE presented the smallest SD values among the AES models for both the proposed and conventional methods. A decreased SD may facilitate increasing agreement in AES predictions when rater assignments in the training data change, but this does not necessarily increase agreement between predicted and gold-standard scores. We can, thus, interpret the different performance of EASE in the two experiments as being due to the small SD. This analysis suggests that a method with high performance in both experiments is better. Tables V and VI suggest that BERT is the best model, because it provided high performance in both experiments.

Table VII shows that the proposed method resulted in smaller SD values than did the conventional method for all AES models. The main reason for this is that the proposed method mitigated the effects of extreme and aberrant ratings. Note that although a decreased SD may increase agreement in the first experiment, as described above, the proposed method provided higher performance than did the conventional method in both experiments. This suggests that the decreased SD in the proposed method, induced by mitigating the rater effects, improves the overall AES performance.

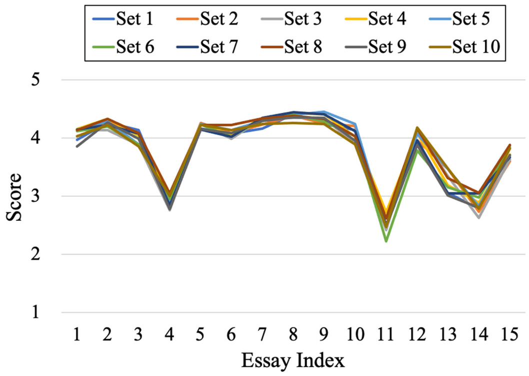


Fig. 8. Predicted scores for 15 essays in the proposed BERT-based models, trained on ten datasets.

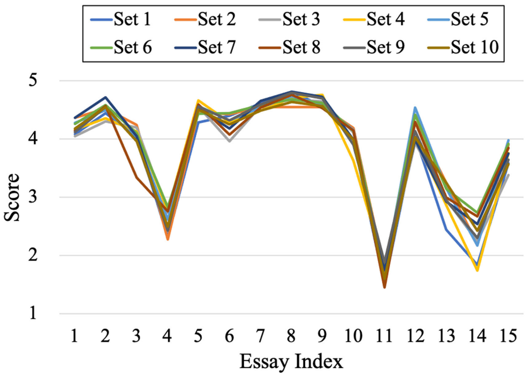


Fig. 9. Predicted scores for 15 essays in the conventional BERT models, trained on ten datasets.

We next analyze in more detail the output scores of the proposed and conventional methods when changing rater assignments in the training dataset. For this analysis, Figs. 8 and 9 show output scores from the proposed and conventional BERT-based models, respectively. Here, we selected the BERT-based model because it showed high performance in both experiments. Figs. 8 and 9 plot output scores for the first 15 essays as obtained using each dataset, with horizontal axes showing the essay index j and vertical axes showing the score value. Set l in the legend indicates correspondence to the lth dataset. These figures indicate that scores output from the conventional method tended to vary when dataset rater assignments changed, whereas the proposed method successfully reduced such fluctuations. This can be confirmed especially for essays 3, 13, and 14 in the figures.

To quantitatively evaluate this point, we calculated the SD of output scores for each essay obtained when using the ten datasets. Concretely, letting $\hat{U}'_{lj}$ be the output score of an AES model for essay e_j when the lth generated dataset $\boldsymbol{U}'_l$ is used, the SD for the jth essay, σ_j, was calculated as the SD of scores $\hat{\boldsymbol{U}}'_j = \{\hat{U}'_{lj} \,|\, l \in \{1, \ldots, 10\}\}$. We, then, calculated the average value of the SD σ_j. The averaged σ_j value $\bar{\sigma}$ becomes large if the output scores for the same essay vary when rater assignments in the training data change. The $\bar{\sigma}$ column of Table VII shows the results, which indicate that the proposed method provides a smaller $\bar{\sigma}$ in all cases. We also applied Student's t-test to compare $\bar{\sigma}$ between the proposed and conventional methods for each base AES model. The results showed that $\bar{\sigma}$ values in the proposed method were significantly smaller than those in the conventional method in all cases ($p < 0.001$). This analysis demonstrates that the proposed method can remove the effects of rater biases, thereby improving the robustness of AES.

VII. Conclusion

In this article, we proposed a new method that trains AES models using IRT-based unbiased scores to mitigate dependence of AES model performance on the characteristics of raters grading essays in training data. Through experiments using an actual dataset, we demonstrated that the proposed method provides more robust essay scores compared with conventional AES models. We also showed that the proposed method improved rating prediction accuracy. The proposed method is simple and easily applicable to various existing AES models, but it effectively improves the performance of AES models. As described in Section I, our method is also highly suited to situations where high-quality training data are hard to prepare, including educational contexts.

In future studies, we plan to evaluate the effectiveness of the proposed method by using various datasets. For example, large-scale experiments using crowdsourcing platforms might provide useful and detailed findings for discussing how reliable crowd workers' ratings are and what kinds of workers we should hire for essay rating tasks. Furthermore, we would like to conduct the same experiments in this article but with essay data on other topics from the ASAP dataset. The fifth ASAP topic this study used was a source-based essay writing task, in which the written essays were relatively short. Compared with topics having a higher degree of freedom or those for which the essays are longer, an accurate essay score prediction is relatively easy for this topic. Meanwhile, the differences in rating behavior among raters might be larger for other such topics because the variety of the subject matter in the essays is greater. We expect that the existence of larger rater biases will further demonstrate the effectiveness of the proposed method.

Another topic for future study is developing an end-to-end training procedure of the proposed method. This study separately trained an IRT model and an AES model. However, end-to-end training is expected to further improve performance, because the IRT-based score θ_j can be more accurately estimated using both rating data and textual essay information.

Implementing a function to check the quality of predicted scores is another future research direction because we are sometimes interested in knowing the reliability or confidence level of scores predicted by AES. Furthermore, taking advantage of the unique property of the proposed method, namely, its high interpretability in terms of rater biases, another future direction is to analyze how rater biases affect the behavior of AES models, for example, by using an explanation model, as in [83].

References

[1] H. J. Bernardin, S. Thomason, M. R. Buckley, and J. S. Kane, "Rater rating-level bias and accuracy in performance appraisals: The impact of rater personality, performance management competence, and rater accountability," *Hum. Resour. Manage.*, vol. 55, no. 2, pp. 321–340, Mar./Apr. 2016, doi: 10.1002/hrm.21678.

[2] M. A. Hussein, H. A. Hassan, and M. Nassef, “Automated language essay scoring systems: A literature review,” *Peer J. Comput. Sci.*, vol. 5, Aug. 2019, Art. no. e208, doi: 10.7717/peerj-cs.208.

[3] O. L. Liu, L. Frankel, and K. C. Roohr, “Assessing critical thinking in higher education: Current state and directions for next-generation assessment,” *ETS Res. Rep. Ser.*, vol. 2014, no. 1, pp. 1–23, Jun. 2014, doi: 10.1002/ets2.12009.

[4] Y. Rosen and M. Tager, “Making student thinking visible through a concept map in computer-based assessment of critical thinking,” *J. Educ. Comput. Res.*, vol. 50, no. 2, pp. 249–270, Mar. 2014, doi: 10.2190/EC.50.2.f.

[5] Y. Abosalem, “Assessment techniques and students’ higher-order thinking skills,” *Int. J. Secondary Educ.*, vol. 4, no. 1, pp. 1–11, Feb. 2016.

[6] Z. Ke and V. Ng, “Automated essay scoring: A survey of the state of the art,” in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Macao, China, Aug. 10–16, 2019, pp. 6300–6308, doi: 10.24963/ijcai.2019/879.

[7] T. Eckes, *Introduction to Many-Facet Rasch Measurement: Analyzing and Evaluating Rater-Mediated Assessments*. Bern, Switzerland: Peter Lang, 2011.

[8] C. Hua and S. A. Wind, “Exploring the psychometric properties of the mind-map scoring rubric,” *Behaviormetrika*, vol. 46, no. 1, pp. 73–99, Apr. 2019, doi: 10.1007/s41237-018-0062-z.

[9] N. L. A. Kassim, “Judging behaviour and rater errors: An application of the many-facet Rasch model,” *GEMA Online J. Lang. Stud.*, vol. 11, no. 3, pp. 179–197, Sep. 2011.

[10] C. M. Myford and E. W. Wolfe, “Understanding Rasch measurement: Detecting and measuring rater effects using many-facet Rasch measurement: Part I,” *J. Appl. Meas.*, vol. 4, no. 4, pp. 386–422, 2003.

[11] C. M. Myford and E. W. Wolfe, “Understanding Rasch measurement: Detecting and measuring rater effects using many-facet Rasch measurement: Part II,” *J. Appl. Meas.*, vol. 5, no. 2, pp. 189–227, 2004.

[12] A. A. Rahman, J. Ahmad, R. M. Yasin, and N. M. Hanafi, “Investigating central tendency in competency assessment of design electronic circuit: Analysis using many facet Rasch measurement (MFRM),” *Int. J. Inf. Educ. Technol.*, vol. 7, no. 7, pp. 525–528, Jul. 2017, doi: 10.18178/ijiet.2017.7.7.923.

[13] M. Uto and M. Ueno, “Empirical comparison of item response theory models with Rater’s parameters,” *Heliyon*, vol. 4, no. 5, pp. 1–32, May 2018, doi: 10.1016/j.heliyon.2018.e00622.

[14] K. Zupanc and Z. Bosnić, “Increasing accuracy of automated essay grading by grouping similar graders,” in *Proc. 8th Int. Conf. Web Intell., Mining Semantics*, Novi Sad, Serbia, Jun. 25–27, 2018, pp. 1–6, doi: 10.1145/3227609.3227645.

[15] E. Amorim, M. Cançado, and A. Veloso, “Automated essay scoring in the presence of biased ratings,” in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, New Orleans, LA, USA, Jun. 1–6, 2018, pp. 229–237, doi: 10.18653/v1/N18-1021.

[16] M. Dascalu, W. Westera, S. Ruseti, S. Trausan-Matu, and H. Kurvers, “ReaderBench learns Dutch: Building a comprehensive automated essay scoring system for Dutch language,” in *Proc. 18th Int. Conf. Artif. Intell. Educ.*, Wuhan, China, Jun. 28–Jul. 2, 2017, pp. 52–63, doi: 10.1007/978-3-319-61425-0_5.

[17] J. C. B. Mark and D. Shermis, *Automated Essay Scoring: A Cross-Disciplinary Perspective*. Evanston, IL, USA: Routledge, 2002.

[18] H. V. Nguyen and D. J. Litman, “Argument mining for improving the automated scoring of persuasive essays,” in *Proc. 32nd AAAI Conf. Artif. Intell.*, New Orleans, LA, USA, Feb. 2–7, 2018, vol. 32, pp. 5892–5899.

[19] D. Alikaniotis, H. Yannakoudakis, and M. Rei, “Automatic text scoring using neural networks,” in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Berlin, Germany, Aug. 7–12, 2016, pp. 715–725, doi: 10.18653/v1/P16-1068.

[20] K. Taghipour and H. T. Ng, “A neural approach to automated essay scoring,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Austin, TX, USA, Nov. 1–5, 2016, pp. 1882–1891, doi: 10.18653/v1/D16-1193.

[21] T. Dasgupta, A. Naskar, L. Dey, and R. Saha, “Augmenting textual qualitative features in deep convolution recurrent neural network for automatic essay scoring,” in *Proc. 5th Workshop Natural Lang. Process. Techn. Educ. Appl.*, Melbourne, VIC, Australia, Jul. 19, 2018, pp. 93–102, doi: 10.18653/v1/W18-3713.

[22] Y. Farag, H. Yannakoudakis, and T. Briscoe, “Neural automated essay scoring and coherence modeling for adversarially crafted input,” in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, New Orleans, LA, USA, Jun. 1–6, 2018, pp. 263–271, doi: 10.18653/v1/N18-1024.

[23] C. Jin, B. He, K. Hui, and L. Sun, “TDNN: A two-stage deep neural network for prompt-independent automated essay scoring,” in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, Melbourne, VIC, Australia, Jul. 15–20, 2018, pp. 1088–1097.

[24] M. Mesgar and M. Strube, “A neural local coherence model for text quality assessment,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Brussels, Belgium, Oct. 31–Nov. 4, 2018, pp. 4328–4339, doi: 10.18653/v1/D18-1464.

[25] Y. Wang, Z. Wei, Y. Zhou, and X. Huang, “Automatic essay scoring incorporating rating schema via reinforcement learning,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Brussels, Belgium, Oct. 31–Nov. 4, 2018, pp. 791–797, doi: 10.18653/v1/D18-1090.

[26] F. S. Mim, N. Inoue, P. Reisert, H. Ouchi, and K. Inui, “Unsupervised learning of discourse-aware text representation for essay scoring,” in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics Student Res. Workshop*, Florence, Italy, Jul. 28–Aug. 2, 2019, pp. 378–385, doi: 10.18653/v1/P19-2053.

[27] F. Nadeem, H. Nguyen, Y. Liu, and M. Ostendorf, “Automated essay scoring with discourse-aware neural models,” in *Proc. 14th Workshop Innov. Use NLP Building Educ. Appl.*, Florence, Italy, Aug. 2, 2019, pp. 484–493, doi: 10.18653/v1/W19-4450.

[28] M. Uto, Y. Xie, and M. Ueno, “Neural automated essay scoring incorporating handcrafted features,” in *Proc. 28th Int. Conf. Comput. Linguistics*, Barcelona, Spain, Dec. 8–13, 2020, pp. 6077–6088, doi: 10.18653/v1/2020.coling-main.535.

[29] R. Ridley, L. He, X.-Y. Dai, S. Huang, and J. Chen, “Automated cross-prompt scoring of essay traits,” in *Proc. 35th. AAAI Conf. Artif. Intell.*, Feb. 1–9, 2021, vol. 35, pp. 13745–13753.

[30] M. Uto, “A review of deep-neural automated essay scoring models,” *Behaviormetrika*, vol. 48, no. 2, pp. 4459–484, Jul. 2021, doi: 10.1007/s41237-021-00142-y.

[31] F. Saal, R. Downey, and M. Lahey, “Rating the ratings: Assessing the psychometric quality of rating data,” *Psychol. Bull.*, vol. 88, no. 2, pp. 413–428, Sep. 1980, doi: 10.1037/0033-2909.88.2.413.

[32] M. Uto, “Rater-effect IRT model integrating supervised LDA for accurate measurement of essay writing ability,” in *Proc. 20th Int. Conf. Artif. Intell. Educ.*, Chicago, IL, USA, Jun. 25–29, 2019, pp. 494–506, doi: 10.1007/978-3-030-23204-7_41.

[33] M. Uto, N. D. Thien, and M. Ueno, “Group optimization to maximize peer assessment accuracy using item response theory,” in *Proc. 18th Int. Conf. Artif. Intell. Educ.*, Wuhan, China, Jun. 28–Jul. 2, 2017, pp. 393–405, doi: 10.1007/978-3-319-61425-0_33.

[34] J. Amidei, P. Piwek, and A. Willis, “Identifying annotator bias: A new IRT-based method for bias identification,” in *Proc. 28th Comput. Linguistics*, Barcelona, Spain, Dec. 8–13, 2020, pp. 4787–4797, doi: 10.18653/v1/2020.coling-main.421.

[35] T. Eckes, “Examining rater effects in TestDaF writing and speaking performance assessments: A many-facet Rasch analysis,” *Lang. Assessment Quart.*, vol. 2, no. 3, pp. 197–221, Nov. 2005, doi: 10.1207/s15434311laq0203_2.

[36] J. Huang, L. Qu, R. Jia, and B. Zhao, “O2U-Net: A simple noisy label detection approach for deep neural networks,” in *Proc. IEEE Int. Conf. Comput. Vis.*, Seoul, South Korea, Oct. 27–Nov. 2, 2019, pp. 3326–3334, doi: 10.1109/ICCV.2019.00342.

[37] S. A. Wind, E. W. Wolfe, G. Engelhard, Jr., P. Foltz, and M. Rosenstein, “The influence of rater effects in training sets on the psychometric quality of automated scoring for writing assessments,” *Int. J. Testing*, vol. 18, no. 1, pp. 27–49, Nov. 2018, doi: 10.1080/15305058.2017.1361426.

[38] S. Li *et al.*, “Coupled-view deep classifier learning from multiple noisy annotators,” in *Proc. 34th AAAI Conf. Artif. Intell.*, New York, NY, USA, Feb. 7–12, 2020, vol. 34, pp. 4667–4674, doi: 10.1609/aaai.v34i04.5898.

[39] R. J. Patz, B. W. Junker, M. S. Johnson, and L. T. Mariano, “The hierarchical rater model for rated test items and its application to large-scale educational assessment data,” *J. Educ. Behav. Statist.*, vol. 27, no. 4, pp. 341–384, Dec. 2002, doi: 10.3102/10769986027004341.

[40] R. J. Patz and B. Junker, “Applications and extensions of MCMC in IRT: Multiple item types, missing data, and rated responses,” *J. Educ. Behav. Statist.*, vol. 24, no. 4, pp. 342–366, Dec. 1999, doi: 10.3102/10769986024004342.

[41] 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.

[42] M. Uto and M. Ueno, “Item response theory without restriction of equal interval scale for rater’s score,” in *Proc. 19th Int. Conf. Artif. Intell. Educ.*, London, U.K., Jun. 25–30, 2018, pp. 363–368, doi: 10.1007/978-3-319-93846-2_68.

[43] M. Uto and M. Ueno, “A generalized many-facet Rasch model and its Bayesian estimation using Hamiltonian Monte Carlo,” *Behaviormetrika*, vol. 47, no. 2, pp. 469–496, May 2020, doi: 10.1007/s41237-020-00115-7.

[44] M. Uto, N. Duc Thien, and M. Ueno, "Group optimization to maximize peer assessment accuracy using item response theory and integer programming," *IEEE Trans. Learn. Technol*, vol. 13, no. 1, pp. 91–106, Feb. 2020, doi: 10.1109/TLT.2019.2896966.

[45] I. Aomi, E. Tsutsumi, M. Uto, and M. Ueno, "Integration of automated essay scoring models using item response theory," in *Proc. 22nd Int. Conf. Artif. Intell. Educ.*, Utrecht, The Netherlands, Jun. 14–18, 2021, pp. 54–59, doi: 10.1007/978-3-030-78270-2_9.

[46] Y. Attali and J. Burstein, "Automated essay scoring with e-rater V.2," *J. Technol. Learn. Assessment*, vol. 4, no. 3, pp. 1–31, Feb. 2006.

[47] P. Phandi, K. M. A. Chai, and H. T. Ng, "Flexible domain adaptation for automated essay scoring using correlated linear regression," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Lisbon, Portugal, Sep. 17–21, 2015, pp. 431–439, doi: 10.18653/v1/D15-1049.

[48] B. Beigman Klebanov, M. Flor, and B. Gyawali, "Topicality-based indices for essay scoring," in *Proc. 11th Workshop Innov. Use NLP Building Educ. Appl.*, San Diego, CA, USA, Jun. 16–17, 2016, pp. 63–72, doi: 10.18653/v1/W16-0507.

[49] M. Cozma, A. Butnaru, and R. T. Ionescu, "Automated essay scoring with string kernels and word embeddings," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, Melbourne, VIC, Australia, Jul. 15–20, 2018, pp. 503–509, doi: 10.18653/v1/P18-2080.

[50] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. 17th Annu. Conf. North Amer. Ch. Assoc. Comput. Linguistics: Hum. Lang. Technol.*, Minneapolis, MN, USA, Jun. 2–7, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.

[51] P. U. Rodriguez, A. Jafari, and C. M. Ormerod, "Language models and automated essay scoring," 2019, *arXiv:1909.09482*.

[52] T. Liu, W. Ding, Z. Wang, J. Tang, G. Y. Huang, and Z. Liu, "Automatic short answer grading via multiway attention networks," in *Proc. 20th Int. Conf. Artif. Intell. Educ.*, Chicago, IL, USA, Jun. 25–29, 2019, pp. 169–173, doi: 10.1007/978-3-030-23207-8_32.

[53] J. Lun, J. Zhu, Y. Tang, and M. Yang, "Multiple data augmentation strategies for improving performance on automatic short answer scoring," in *Proc. 34th AAAI Conf. Artif. Intell.*, New York, NY, USA, Feb. 7–12, 2020, vol. 34, pp. 13389–13396, doi: 10.1609/aaai.v34i09.7062.

[54] C. Sung, T. I. Dhamecha, and N. Mukhi, "Improving short answer grading using transformer-based pre-training," in *Proc. 20th Int. Conf. Artif. Intell. Educ.*, Chicago, IL, USA, Jun. 25–29, 2019, pp. 469–481, doi: 10.1007/978-3-030-23204-7_39.

[55] A. Vaswani *et al.*, "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Long Beach, CA, USA, Dec. 4–7, 2017, pp. 5998–6008.

[56] F. Lord, *Applications of Item Response Theory to Practical Testing Problems*. Evanston, IL, USA: Routledge, 1980, doi: 10.4324/9780203056615.

[57] E. Muraki, "A generalized partial credit model," in *Handbook of Modern Item Response Theory*, W. J. van der Linden and R. K. Hambleton, Eds. New York, NY, USA: Springer, 1997, pp. 153–164, doi: 10.1007/978-1-4757-2691-6_9.

[58] M. Uto, "A multidimensional generalized many-facet Rasch model for rubric-based performance assessment," *Behaviormetrika*, vol. 48, no. 2, pp. 425–457, Jul. 2021, doi: 10.1007/s41237-021-00144-w.

[59] H. Suen, "Peer assessment for massive open online courses (MOOCs)," *Int. Rev. Res. Open Distrib. Learn.*, vol. 15, no. 3, pp. 313–327, Jul. 2014, doi: 10.19173/irrodl.v15i3.1680.

[60] N. B. Shah, J. Bradley, S. Balakrishnan, A. Parekh, K. Ramchandran, and M. J. Wainwright, "Some scaling laws for MOOC assessments," in *Proc. ACM KDD Workshop Data Mining Educ. Assessment Feedback*, New York, NY, USA, Aug. 24–27, 2014. [Online]. Available: https://www.stat.cmu.edu/~siva/Papers/MOOC14.pdf

[61] T. Nguyen, M. Uto, Y. Abe, and M. Ueno, "Reliable peer assessment for team project based learning using item response theory," in *Proc. 23rd Int. Conf. Comput. Educ.*, Hangzhou, China, Nov. 30–Dec. 4, 2015, pp. 144–153.

[62] E. W. Wolfe, B. C. Moulder, and C. M. Myford, "Detecting differential rater functioning over time (DRIFT) using a Rasch multi-faceted rating scale model," *J. Appl. Meas.*, vol. 2, no. 3, pp. 256–280, 2001.

[63] S. A. Wind and W. Guo, "Exploring the combined effects of rater misfit and differential rater functioning in performance assessments," *Educ. Psychol. Meas.*, vol. 79, no. 5, pp. 962–987, Oct. 2019, doi: 10.1177/0013164419834613.

[64] F. Baker and S. H. Kim, *Item Response Theory: Parameter Estimation Techniques*. Boca Raton, FL, USA: CRC Press, 2004, doi: 10.1201/9781482276725.

[65] J.-P. Fox, *Bayesian Item Response Modeling: Theory and Applications*. New York, NY, USA: Springer, 2010.

[66] M. D. Hoffman and A. Gelman, "The No-U-Turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1593–1623, Jan. 2014, doi: 10.5555/2627435.2638586.

[67] S. Brooks, A. Gelman, G. Jones, and X. Meng, *Handbook of Markov Chain Monte Carlo*. Boca Raton, FL, USA: CRC Press, 2011.

[68] B. Carpenter *et al.*, "Stan: A probabilistic programming language," *J. Statist. Softw.*, vol. 76, no. 1, pp. 1–32, Jan. 2017, doi: 10.18637/jss.v076.i01.

[69] J. Guo, J. Gabry, B. Goodrich, and S. Weber, "RStan: The R interface to Stan," [Online]. Available: https://mc-stan.org/rstan/

[70] H. J. Shin, S. Rabe-Hesketh, and M. Wilson, "Trifactor models for multiple-ratings data," *Multivariate Behav. Res.*, vol. 54, no. 3, pp. 360–381, Mar. 2019, doi: 10.1080/00273171.2018.1530091.

[71] M. Uto, "Accuracy of performance-test linking based on a many-facet Rasch model," *Behav. Res. Methods*, vol. 53, pp. 1440–1454, Nov. 2021, doi: 10.3758/s13428-020-01498-x.

[72] S. A. Wind and E. Jones, "The effects of incomplete rating designs in combination with rater effects," *J. Educ. Meas.*, vol. 56, no. 1, pp. 76–100, Mar. 2019, doi: 10.1111/jedm.12201.

[73] J. M. Linacre, *Many-Faceted Rasch Measurement*. Chicago, IL, USA: MESA Press, 1989.

[74] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Autom. Control*, vol. AC-19, no. 6, pp. 716–723, Dec. 1974, doi: 10.1109/TAC.1974.1100705.

[75] G. Schwarz, "Estimating the dimensions of a model," *Ann. Statist.*, vol. 6, no. 2, pp. 461–464, Mar. 1978, doi: 10.1214/aos/1176344136.

[76] S. Watanabe, "Asymptotic equivalence of Bayes cross validation and widely applicable information criterion in singular learning theory," *J. Mach. Learn. Res.*, vol. 11, no. 116, pp. 3571–3594, Dec. 2010.

[77] S. Watanabe, "A widely applicable Bayesian information criterion," *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 867–897, Mar. 2013.

[78] A. Gelman, J. Carlin, H. Stern, D. Dunson, A. Vehtari, and D. Rubin, *Bayesian Data Analysis*, 3rd ed. Boca Raton, FL, USA: CRC Press, 2013.

[79] W. J. van der Linden, "Handbook of item response theory," in *Statistical Tools*, vol. 2. Boca Raton, FL, USA: CRC Press, 2016.

[80] M. L. Nering and R. Ostini, *Handbook of Polytomous Item Response Theory Models*. Evanston, IL, USA: Routledge, 2010.

[81] T. D. Tran, "Bayesian analysis of multivariate longitudinal data using latent structures with applications to medical data," Ph.D. dissertation, Dept. Public Health Primary Care, KU Leuven, Belgium, 2020.

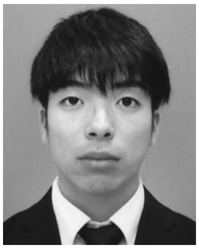
[82] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Dec. 2014, *arXiv:1412.6980*.

[83] V. Kumar and D. Boulanger, "Explainable automated essay scoring: Deep learning really has pedagogical value," *Front. Educ.*, vol. 5, Oct. 2020, Art. no. 572367, doi: 10.3389/feduc.2020.572367.

Masaki Uto received the Ph.D. degree in engineering from the University of Electro-Communications (UEC), Chofu, Japan, in 2013.

He has been an Associate Professor with the Graduate School of Informatics and Engineering, UEC, since 2020. His research interests include educational and psychological measurement, Bayesian statistics, machine learning, and natural language processing.

Dr. Uto was the recipient of the Best Paper Runner-Up Award at the 2020 International Conference on Artificial Intelligence in Education.



Masashi Okano received the B.S. degree in engineering, from the University of Electro-Communications, Chofu, Japan, in 2020, where he is currently working toward the M.S. degree in engineering.

His research interests include educational and psychological measurement, machine learning, and natural language processing.

Mr. Okano was the recipient of the Best Paper Runner-Up Award at the 2020 International Conference on Artificial Intelligence in Education.