Preventing Critical Scoring Errors in Short Answer Scoring with Confidence Estimation

Hiroaki Funayama[1,2] **Shota Sasaki**[2,1] **Yuichiroh Matsubayashi**[1,2]
Tomoya Mizumoto[3,2] **Jun Suzuki**[1,2] **Masato Mita**[2,1] **Kentaro Inui**[1,2]
[1] Tohoku University [2] RIKEN [3] Future Corporation
{hiroaki, jun.suzuki, inui}@ecei.tohoku.ac.jp
{shota.sasaki.yv, tomoya.mizumoto, masato.mita}@riken.jp
{y.m}@tohoku.ac.jp

Abstract

Many recent Short Answer Scoring (SAS) systems have employed Quadratic Weighted Kappa (QWK) as the evaluation measure of their systems. However, we hypothesize that QWK is unsatisfactory for the evaluation of the SAS systems when we consider measuring their effectiveness in actual usage. We introduce a new task formulation of SAS that matches the actual usage. In our formulation, the SAS systems should extract as many scoring predictions that are not *critical scoring errors* (CSEs). We conduct the experiments in our new task formulation and demonstrate that a typical SAS system can predict scores with *zero CSE* for approximately 50% of test data at maximum by filtering out low-reliablility predictions on the basis of a certain confidence estimation. This result directly indicates the possibility of reducing half the scoring cost of human raters, which is more preferable for the evaluation of SAS systems.

1 Introduction

The automated Short Answer Scoring (SAS) is a task of estimating a score of a short-text answer written as a response to a given prompt on the basis of whether the answer satisfies the rubrics prepared by a human in advance. SAS systems have mainly been developed to markedly reduce the scoring cost of human raters. Moreover, the SAS systems play a central role in providing stable and sustainable scoring in a repeated and large-scale examination and (online) self-study learning support system (Attali and Burstein, 2006; Shermis et al., 2010; Leacock and Chodorow, 2003; Burrows et al., 2015).

The development of the SAS systems has a long history (Page, 1994; Foltz et al., 1999). Many recent previous studies, e.g., (Mizumoto et al., 2019; Taghipour and Ng, 2016; Riordan et al., 2017; Wang et al., 2019), utilize Quadratic Weighted Kappa (QWK) (Cohen, 1968) as a measure for the achievement and for the comparison of the performances of the SAS systems. QWK is indeed useful for measuring and comparing the overall performance of each system and the daily developments of their scoring models. In our experiments, however, we reveal that the SAS systems with high QWK potentially incur serious scoring errors (see experiments in Section 5.3). Such serious scoring errors are rarely incurred by trained human raters, therefore, we need to avoid containing this type of errors to ensure the sufficient scoring quality, for use in the scoring of commercial examinations, of SAS systems. When we strictly focus on measuring the effectiveness of the SAS systems in actual usage, QWK seems unsatisfactory for the evaluation of the SAS systems. Here, we assume that the following procedure is a realistic configuration for utilizing the SAS systems in actual usage: (1) apply a SAS system to score each answer, (2) treat the predicted score as the final decision if the predicted score is highly reliable, proceed to the next step otherwise, and (3) discard the unreliable predicted score and reevaluate the answer by a human rater as the final decision. Therefore, we aim to establish an appropriate evaluation scheme for accurately estimating the effectiveness of the SAS systems in actual usage instead of the current de facto standard evaluation measure, QWK.

To do so, we first introduce a key concept **critical scoring error (CSE)**, which reflects unacceptable prediction error. Specifically, CSE refers to the observation that the gap between a predicted score and the ground truth is larger than a predefined threshold, which, for example, can be determined by an average gap in human raters. Then, in our task formulation, the goal of the automated SAS is to obtain as many predictions without CSE as possible, which directly reflects the effectiveness of the SAS models in the actual usage. We also in-

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pages 237–243

Prompt

Explain what the author means by the phrase "this tension has caused several different philosophical viewpoints in Western culture"

Student Response

Conflicts of interest in Western culture are formed on the basis of God vs. Human, … **Score: 4 points**

Figure 1: Example of a prompt and a student's short-text response excerpted from the dataset proposed by (Mizumoto et al., 2019). The allotment score of this prompt is 16, and this response is assigned four points by a human rater. Note that the prompt and the response are translated from the original ones given in Japanese.

troduce the **critical scoring error rate (CSRate)**, which is the CSE rate in a subset of the test data selected on the basis of the confidence measure of predictions, for evaluating the performance of the SAS systems.

In our experiments, we select two methods, i.e., posterior probability and *trust score* (Jiang et al., 2018), as case studies of estimating whether or not each prediction is reliable. We use those two confidence estimation methods to obtain a set of highly reliable predictions. The experimental results show that the SAS systems can predict scores with *zero CSE* for approximately 50% of test data at maximum by filtering low-reliability predictions.

2 Short Answer Scoring

2.1 Task Description

As an example, in Figure 1, for a short answer question, a student writes a short text as a response to a given prompt. A human rater marks the response on the basis of the rubrics for the prompt. Similarly, given a student response $\mathbf{x} = \{x_1, x_2, ..., x_n\}$ for a prompt allotted N points, our short answer scoring task can be defined as predicting a score of $s \in C = \{0, ..., N\}$ for that response.

SAS models are often evaluated in terms of the agreement between the scores of a model prediction and human annotation with QWK. QWK is calculated as:

$$\kappa = 1 - \frac{\sum_{i,j} \mathbf{W}_{i,j}\mathbf{O}_{i,j}}{\sum_{i,j} \mathbf{W}_{i,j}\mathbf{E}_{i,j}}, \quad (1)$$

where $\mathbf{O} \in \mathbb{R}^{N \times N}$ is the confusion matrix of two ratings and $\mathbf{E} \in \mathbb{R}^{N \times N}$ is the outer product of histogram vectors of the two ratings; $\mathbf{O}$ and $\mathbf{E}$ are normalized to have the same sum of their elements. $\mathbf{W}_{i,j}$ is calculated as:

$$\mathbf{W}_{i,j} = \frac{(i-j)^2}{(N-1)^2}, \quad (2)$$

where i and j are the score rated by a human and the score predicted by a SAS system, respectively. N is allotment score defined for a prompt.

2.2 Scoring Model

Following related works (Nguyen and O'Connor, 2015; Jiang et al., 2018; Hendrycks and Gimpel, 2017) on confidence calibration, we formalize our SAS model as a classification model. Note that our focus in this paper is more on evaluating the effectiveness of the confidential scores on SAS tasks than on creating an accurate SAS model. Therefore, we employ a standard Bidirectional Long Short Term Memory (Bi-LSTM) based neural network for our scoring model as a representative model for typical SAS tasks.

Given an input student response $\mathbf{x}$, the model outputs a score $s \in \mathcal{S}$ for the response as follows. First, we convert tokens in $\mathbf{x}$ to word-embedding vectors. These embeddings are fed into a Bi-LSTM and D dimensional hidden vectors $\{\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n\}$ are obtained as the sum of the hidden vectors from forward and backward LSTMs. The response vector $\widetilde{\mathbf{h}}$ is then computed by averaging these hidden vectors.

$$\widetilde{\mathbf{h}} = \frac{1}{n}\sum_{t=1}^{n} \mathbf{h}_t \quad (3)$$

A probability distribution of the score is calculated as:

$$p(s|\mathbf{x}) = \texttt{softmax}(\mathbf{W}\widetilde{\mathbf{h}} + \mathbf{b}), \quad (4)$$

where $\mathbf{W} \in \mathbb{R}^{N \times D}$ and $\mathbf{b} \in \mathbb{R}^{N}$ are learnable parameters. Finally, we select the most likely output score $\widehat{s} \in \mathcal{S}$ for given input $\mathbf{x}$ as:

$$\widehat{s} = \arg\max_{s \in \mathcal{S}} \{p(s|\mathbf{x})\}. \quad (5)$$

3 Task Formulation

The goal in our new task formulation for applying SAS to real-world educational measurements is to obtain as many scoring predictions without CSEs as possible. This is because we can trust such predictions and markedly reduce the cost of the human scoring effort. In this section, we describe our new task formulation of the automated SAS.

	Prompt 1	Prompt 2	Prompt 3	Prompt 4	Prompt 5	Prompt 6
Length limit (char.)	70	50	60	70	70	60
Average score	7.40	4.43	5.73	5.78	4.81	5.70
Allotment score ($= N$)	16	12	12	15	15	14
Human agreement	.96 (.93)	.94 (.92)	.76 (.79)	.84 (.70)	.82 (.83)	.90 (.82)

Table 1: Statistics of the dataset used in this paper. "Length limit (char.)" is the maximum character length of the response permitted for a prompt. "Allotment score" is the maximum score for a prompt. "Human agreement" represents QWK and Cohen's Kappa (shown in brackets) between the scores annotated by two human raters.

First, to evaluate the proportion of CSEs in the predictions, we define a function on the gold dataset $\mathcal{D}$ that returns whether or not the predicted score $\widehat{s}$ of an input $\mathbf{x}$ is categorized as a CSE:

$$\mathrm{CSE}(\mathbf{x}, s) = \begin{cases} 1 & \text{if} \quad |s - \widehat{s}| \geq \lambda \cdot N \\ 0 & \text{otherwise} \end{cases}, \quad (6)$$

where $\lambda \in [0, 1]$ is a given threshold, N is the allotment of a score for a prompt, s is the ground truth score of input $\mathbf{x}$, and $\widehat{s}$ is obtained using Equation 5. Note that we can choose the value of λ depending on the situation. For example, for an important examination such as an entrance examination, λ should be smaller than that for daily tests in schools.

Here, let $\mathcal{D}$ be a test data set. Moreover, let $\mathcal{D}'$ be a subset of $\mathcal{D}$, that is, $\mathcal{D}' \subseteq \mathcal{D}$. Then, our objective is to maximize the size of the subset $\mathcal{D}'$ on the condition that this subset does not contain CSEs. For obtaining $\mathcal{D}'$, we estimate a confidence score $C(\mathbf{x}, \widehat{s})$ for each prediction on the basis of a certain confidence measure, and then gather the predictions with high confidence scores that exceed a threshold, τ. Therefore, for the evaluation of the performance of our task formulation, we propose a *critical scoring error rate* (CSRate) defined as:

$$\mathrm{CSRate}(\mathcal{D}, \tau) = \frac{1}{|\mathcal{D}'|} \sum_{(\mathbf{x}, s) \in \mathcal{D}'} \mathrm{CSE}(\mathbf{x}, s), \quad (7)$$

$$\mathcal{D}' = \{(\mathbf{x}, s) \in \mathcal{D} | C(\mathbf{x}, \widehat{s}) \geq \tau\}, \quad (8)$$

where $\widehat{s}$ is obtained using Equation 5. In real-world tasks, the model is expected to select as large a subset D' as possible with very small or ideally zero CSRate.

4 Filtering Out Low-Reliability Estimation Using Confidence Score

As described in Equation 8, the quality of the confidence measure is important for our task configuration. In this paper, we employ two methods for computing the confidence score: (1) *posterior probability* of the classification model and (2) *trust score* (Jiang et al., 2018) as case studies.

4.1 Posterior Probability

The most straightforward method for computing the confidence of the prediction in a classification problem is to employ a probabilistic model and use the output label probability:

$$C_{\mathrm{prob}}(\mathbf{x}, \widehat{s}) = p(\widehat{s}|\mathbf{x}). \quad (9)$$

Although a label probability is often used as a confidence score for prediction, some authors are skeptical of its utility (Guo et al., 2017; Kumar et al., 2018). In our experiments, we evaluate the effectiveness of this posterior probability in terms of a confidence estimation method for SAS models.

4.2 Trust Score

Trust score (Jiang et al., 2018) is an indicator of the reliability of prediction based on the distance between a target data point and its nearest data points in training data. The intuition behind this score is that the reliability of a prediction is higher when the target data point is closer to the nearest training data point with the same label and farther away from the nearest training data point with a different label.

In this paper, trust score is calculated as follows. Given a training data value $\{(\mathbf{x}_1, s_1), ..., (\mathbf{x}_m, s_m)\}$, a target data point $\mathbf{x}$ for prediction, and its predicted label $\widehat{s}$, we first obtain a vector representation for each data point. In our model, the representation for each data point is the sentence vector of the student response described in Section 2.2. Let $\mathcal{H} = \{\widetilde{\mathbf{h}}_1, ..., \widetilde{\mathbf{h}}_m\}$ be a set of vector representations for the training data points and let $\widetilde{\mathbf{h}}_{\mathbf{x}}$ be a vector for the target data point $\mathbf{x}$. Then we collect the representations in the training data that have the same label as the predicted label $\widehat{s}$:

$$\mathcal{H}_{\widehat{s}} = \{\widetilde{\mathbf{h}}_k \in \mathcal{H} \mid s_k = \widehat{s}\}. \quad (10)$$

The trust score C_{trust} for $\mathbf{x}$ is then calculated as the ratio of the euclidean distance $d(\cdot, \cdot)$ between the target representation $\widetilde{\mathbf{h}}_{\mathbf{x}}$ and two data-point representations $\widetilde{\mathbf{h}}_p$ and $\widetilde{\mathbf{h}}_c$ in the training data:

$$C_{\text{trust}}(\mathbf{x}, \widehat{s}, \mathcal{H}) = \frac{d(\widetilde{\mathbf{h}}_{\mathbf{x}}, \widetilde{\mathbf{h}}_c)}{d(\widetilde{\mathbf{h}}_{\mathbf{x}}, \widetilde{\mathbf{h}}_p) + d(\widetilde{\mathbf{h}}_{\mathbf{x}}, \widetilde{\mathbf{h}}_c)}, \quad (11)$$

where, $\widetilde{\mathbf{h}}_p$ is the representation of the nearest training data point having the same label as the predicted label $\widehat{s}$, and $\widetilde{\mathbf{h}}_c$ is the nearest training data point with a different label:

$$\widetilde{\mathbf{h}}_p = \underset{\widetilde{\mathbf{h}} \in \mathcal{H}_{\widehat{s}}}{\arg\min}\, d(\widetilde{\mathbf{h}}_{\mathbf{x}}, \widetilde{\mathbf{h}}), \quad (12)$$

$$\widetilde{\mathbf{h}}_c = \underset{\widetilde{\mathbf{h}} \in (\mathcal{H} \setminus \mathcal{H}_{\widehat{s}})}{\arg\min}\, d(\widetilde{\mathbf{h}}_{\mathbf{x}}, \widetilde{\mathbf{h}}). \quad (13)$$

5 Experiments

5.1 Dataset

We use the Japanese short answer scoring dataset[1] introduced by Mizumoto et al. (2019). The dataset consists of six prompts. Each prompt has its rubric, student responses, and scores. The prompts, rubrics, and student responses in the dataset were collected from the examinations conducted by a Japanese education company, Takamiya Gakuen Yoyogi Seminar. Each response was manually scored using the multiple analytic criteria for the prompt, and the subscore for each criterion was rated individually on the basis of the corresponding rubric. In the experiments, we use the sum of these analytic scores as a ground truth score of each response.[2]

Table 1 shows the statistics of the dataset. In the dataset, the randomly sampled 100 responses per prompt are annotated by two human raters. Therefore, we can calculate QWKs and their Kappa values (Cohen, 1960) between the two human raters to confirm the degree of human agreement. The Kappa values on this dataset are comparable to or higher than those on other datasets for the SAS task (Leacock and Chodorow, 2003; Mohler and Mihalcea, 2009; Mohler et al., 2011; Basu et al., 2013).

As additional statistics, we calculated the number of CSEs and CSRate in various settings of λ in Equation 6 over the annotated scores of two human raters. Table 2 shows the result. The number of CSEs in Table 2 represents sum of the number of CSEs for all prompts.

λ	#CSEs	CSRate[%]
0.05	171	28.5
0.10	93	15.5
0.15	50	8.33
0.20	38	6.33
0.25	23	3.83
0.30	7	1.17

Table 2: Changes in the number of CSEs and CSRate of two human raters with λ. 100 responses per prompt are graded by two human raters, and the number of CSEs represents the sum of the number of CSEs of each prompt.

5.2 Settings

We split the dataset into training data $(1,600)$, validation data (200), and test data (200). We used pretrained BERT (Devlin et al., 2019) as the embedding layer of the model.[3] We adopted the same optimization algorithm, learning rate, batch size, and output dimension of the recurrent layer as in Taghipour and Ng (2016). We trained the SAS models for 50 epochs and selected the parameters in the epoch in which the best QWK was achieved for the development set. We trained five models with different random seeds and reported the average of the results.

Choosing a reasonable λ that defines CSE is crucial for our formulation. In our experiments, we employed 0.2 as λ for CSE. There is no theoretical and statistical evidence that 0.2 is the optimal value for our formulation. However, as shown in Table 2, 0.2 is assumed to be strict considering that even for human raters make CSEs in about 6% of responses. Therefore, this selection can offer meaningful evaluations for our formulation.

5.3 Result

Can confidence scores filter out CSEs? Figure 2 shows *CSRate*s on test data when we choose a certain proportion of the predicted instances in descending order of the confidence scores. The figure illustrates that the *CSRate* in each prompt increases

[1] https://aip-nlu.gitlab.io/resources/sas-japanese

[2] We ignored the globally subtracted points (e.g., subtraction for spelling errors and omissions) that are originally annotated in the dataset.

[3] We adopted pretrained character-based BERT which is known to be suitable for processing Japanese texts. This is available at https://github.com/cl-tohoku/bert-japanese.

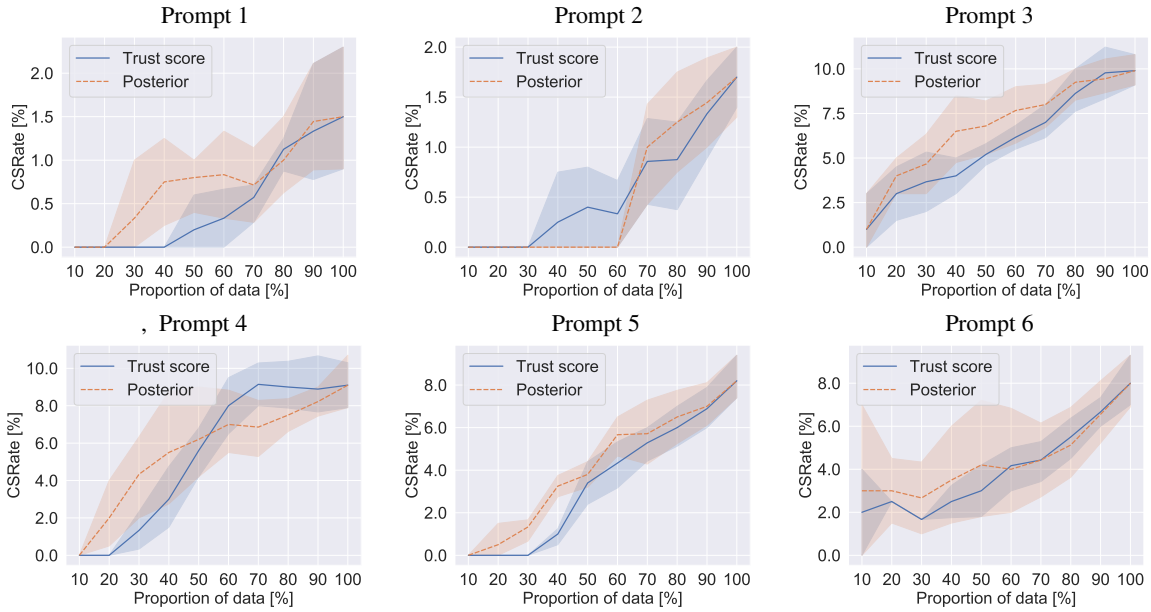

Figure 2: CSRate for test data subsets with the highest confidence scores. Proportion of data represents the ratio of $|\mathcal{D}'|$ to $|\mathcal{D}|$. The lines represent average CSRates of the trained five models and the band represents the maximum and minimum CSRates.

Prompt	Trust Score			Posteior		
	Prop[%]	#CSEs	τ	Prop[%]	#CSEs	τ
Prpt. 1	47.8	0.0	0.56	58.5	1.4	0.86
Prpt. 2	52.0	0.4	0.60	62.2	0.4	0.97
Prpt. 3	14.4	0.2	0.54	27.4	2.6	0.87
Prpt. 4	32.5	0.8	0.54	17.0	0.6	0.93
Prpt. 5	27.6	0.0	0.60	1.5	0.0	1.00
Prpt. 6	27.1	1.2	0.55	21.6	0.6	0.95

Table 3: Proportion of data (Prop[%]) and the number of CSEs (#CSEs) when using the trust score or the posterior probability to filter out unreliable predictions in test data with a certain threshold τ determined by the development set.

	10%		30%		50%		100%
	TS.	Pos.	TS.	Pos.	TS.	Pos.	(Base)
Prpt. 1	1.0	.99	.99	.98	.99	.98	.95
Prpt. 2	1.0	1.0	1.0	1.0	.99	.98	.93
Prpt. 3	.93	.84	.83	.78	.77	.74	.67
Prpt. 4	1.0	1.0	.98	.96	.93	.92	.86
Prpt. 5	1.0	1.0	1.0	.94	.91	.88	.82
Prpt. 6	.94	.92	.95	.95	.93	.92	.88

Table 4: QWK for highly confident predictions. 10%, 30%, and 50% represent the proportion of data with the highest trust score (TS.) or posterior (Pos.). The base represents our model performance on a whole test data.

almost monotonically for both confidence metrics. We can also observe that the CSRate values on four out of six prompts are suppressed to 0% with a certain amount of high confidence predictions (20% to 60% of the test data). This is an important observation for our objective; the result demonstrates that the proposed procedure using confidence scoring possibly obtains a reasonable size of highly reliable predictions. When comparing the two confidence estimation methods, the trust score is more effective for suppressing *CSE*s than the posterior probability on Prompt 1, 3, 4, 5, and 6.

Filtering CSE using the threshold In a practical situation, it is necessary to determine a certain threshold τ in the development set and use it for filtering low- reliability predictions of unknown samples. Assuming this situation, we evaluate how much CSEs in the test set can be reduced by using the threshold τ determined by the procedure described in Section 3.

Table 3 shows the proportions of the remaining test data and the number of CSEs after filtering out low-reliability predictions using the thresholds in each prompt. The results for both confidence estimation methods indicate that we can successfully filter out the unreliable predictions and achieve a sufficiently low CSRate by the proposed approach.

QWK in highly reliable predictions Additionally, we also show QWK of the top 10, 30, and 50%

confident predictions to illustrate the model performance with the de facto standard metric in Table 4. We show QWK of our model predictions on all test data as Base. The table shows that the proposed approach of selecting high-confidence predictions on the basis of confidence scores increases QWK markedly compared with using the whole test data. Moreover, we can achieve a QWK score of 1.0 in some prompts with the top 30% confident predictions, meaning that the model predictions perfectly agree with the ground truth scores.

Note that a higher QWK value does not always mean that the predictions do not contain CSEs. For example, in Table 4, the QWK values for prompts 1 and 2 are higher than 0.9. However, as shown in Figure 2, even with such high QWK values, these predictions include 1.5 to 2.0% of *CSEs*. This observation justifies the concept of CSE. QWK possibly conceals serious mispredictions, which are important to filter out in actual usage.

6 Conclusion and Future Work

In this paper, we introduced a new formulation of the SAS task to evaluate the effectiveness of the SAS systems in actual usage. We defined the concept of a critical scoring error (CSE), which represents unacceptable prediction errors. Then, we formulate the objective of the task to obtain as many predictions without CSE as possible. The experimental results show that by using our proposed procedure of selecting reliable predictions, SAS systems can predict scores with zero CSE for approximately 50% of test data at maximum. This result directly indicates the possibility of reducing half scoring cost of human raters, which, we believe, is highly preferable for the evaluation of SAS systems.

Our study revealed some potential for a better task formulation of SAS that links to actual usage. However, some issues remain, for example, how to determine the effective threshold τ that can strictly guarantee zero CSE is still unknown. This is one major challenge regarding our formulation. Moreover, we must develop a method for more accurately estimating the confidence scores, which is our primary focus in the next step.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP 19H04162 and 19K12112. This work was also partially supported by Bilateral Joint Research Program between RIKEN AIP Center and Tohoku University. We would like to thank the anonymous reviewers for their insightful comments. We also appreciate Takamiya Gakuen Yoyogi Seminar for providing the data.

References

Yigal Attali and Jill Burstein. 2006. Automated Essay Scoring with E-rater v.2.0. *Journal of Technology, Learning, and Assessment*, 4(3):31.

Sumit Basu, Chuck Jacobs, and Lucy Vanderwende. 2013. Powergrading: a Clustering Approach to Amplify Human Effort for Short Answer Grading. *Transactions of the Association for Computational Linguistics*, 1:391–402.

Steven Burrows, Iryna Gurevych, and Benno Stein. 2015. The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, 25(1):60–117.

Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Jacob Cohen. 1968. Weighted Kappa: Nominal Scale Agreement with Provision for Scaled Disagreement or Partial Credit. *Psychological bulletin*, 70(4):213–220.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Peter Foltz, Darrell Laham, and T. Landauer. 1999. The Intelligent Essay Assessor: Applications to Educational Technology. *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, 1(2):939–944.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On Calibration of Modern Neural Networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1321–1330.

Dan Hendrycks and Kevin Gimpel. 2017. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In *Proceedings of 5th International Conference on Learning Representations*.

Heinrich Jiang, Been Kim, Melody Y. Guan, and Maya R. Gupta. 2018. To Trust Or Not To Trust A Classifier. In *Proceedings of Advances in Neural Information Processing Systems 31*, pages 5546–5557.

Aviral Kumar, Sunita Sarawagi, and Ujjwal Jain. 2018. Trainable calibration measures for neural networks from kernel mean embeddings. In *Proceedings of the 35th International Conference on Machine Learning*, pages 2805–2814.

Claudia Leacock and Martin Chodorow. 2003. C-rater: Automated Scoring of Short-Answer Questions. *Computers and the Humanities*, 37(4):389–405.

Tomoya Mizumoto, Hiroki Ouchi, Yoriko Isobe, Paul Reisert, Ryo Nagata, Satoshi Sekine, and Kentaro Inui. 2019. Analytic Score Prediction and Justification Identification in Automated Short Answer Scoring. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 316–325.

Michael Mohler, Razvan Bunescu, and Rada Mihalcea. 2011. Learning to Grade Short Answer Questions using Semantic Similarity Measures and Dependency Graph Alignments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 752–762.

Michael Mohler and Rada Mihalcea. 2009. Text-to-text Semantic Similarity for Automatic Short Answer Grading. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 567–575.

Khanh Nguyen and Brendan O'Connor. 2015. Posterior calibration and exploratory analysis for natural language processing models. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1587–1598.

Ellis Batten Page. 1994. Computer Grading of Student Prose, Using Modern Concepts and Software. *Journal of Experimental Education*, 62(2):127–142.

Brian Riordan, Andrea Horbach, Aoife Cahill, Torsten Zesch, and Chong Min Lee. 2017. Investigating neural architectures for short answer scoring. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 159–168.

M.D. Shermis, J. Burstein, D. Higgins, and Klaus Zechner. 2010. Automated essay scoring: Writing assessment and instruction. *International Encyclopedia of Education*, 4(1):20–26.

Kaveh Taghipour and Hwee Tou Ng. 2016. A Neural Approach to Automated Essay Scoring. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1882–1891.

Tianqi Wang, Naoya Inoue, Hiroki Ouchi, Tomoya Mizumoto, and Kentaro Inui. 2019. Inject Rubrics into Short Answer Grading System. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP*, pages 175–182.