International Journal of Artificial Intelligence in Education (2023) 33:467–496
https://doi.org/10.1007/s40593-022-00294-2

ARTICLE

Automated Short Answer Scoring Using an Ensemble of Neural Networks and Latent Semantic Analysis Classifiers

Christopher Ormerod[1] · Susan Lottridge[1] · Amy E. Harris[1] · Milan Patel[1] · Paul van Wamelen[1] · Balaji Kodeswaran[1] · Sharon Woolf[1] · Mackenzie Young[1]

Accepted: 2 May 2022 / Published online: 2 June 2022

Abstract

We introduce a short answer scoring engine made up of an ensemble of deep neural networks and a Latent Semantic Analysis-based model to score short constructed responses for a large suite of questions from a national assessment program. We evaluate the performance of the engine and show that the engine achieves above-human-level performance on a large set of items. Items are scored using 2-point and 3-point holistic rubrics. We outline the items, data, handscoring methods, engine, and results. We also provide an overview of performance key student groups including: gender, ethnicity, English language proficiency, disability status, and economically disadvantaged status.

Keywords Neural networks · Short answer · Education · Artificial intelligence · Assessment

✉ Christopher Ormerod
Christopher.ormerod@cambiumassessment.com

Susan Lottridge
susan.lottridge@cambiumassessment.com

Amy E. Harris
aharris.air@gmail.com

Milan Patel
milan.patel@cambiumassessment.com

Paul van Wamelen
paul.vanwamelen@cambiumassessment.com

Balaji Kodeswaran
balaji.koedswaran@cambiumassessmnet.com

Sharon Woolf
sherriwoolf@gmail.com

Mackenzie Young
mackenzie.young@cambiumassessment.com

[1] Cambium Assessment, Cambium Learning® Group, Cambium Assessment, Inc. 1000 Thomas Jefferson St., N.W., Washington, DC 20007, USA

Introduction

A comprehensive testing program uses a diverse range of item types, with each type intended to appropriately assess content standards (Darling-Hammond, 2013). Many standards can be assessed using multiple-choice items. However, some standards are best assessed by items in which students are free to construct their own responses. These items can be technology-based, whereby students manipulate controls to answer a question (e.g., draw a line on a graph) or can be text-based, where students can type their answer in a manner that is constrained (e.g., limited key entry) or unconstrained (e.g., unlimited keyboard entry).

The automated assessment of short constructed text responses not only presents a technical challenge, but it also addresses a practical need in educational assessment. Many past studies on the automated scoring of short constructed responses have suggested that automated scoring engines do not meet standards required for operational use (McGraw-Hill Education, 2014). Given that there are a range of methods available to model constructed text responses, it is still an open research problem as to which approach is best suited to which dataset. The aim of this research is to present the results of a neural network-based automated scoring engine that can meet the requirements for operational use in a wide-scale assessment program on a set of short answer items in which the examinees provide unconstrained responses. This includes the specification of a single engine that can score a wide variety of responses, designing rules for flagging inappropriate responses, and a detailed consideration of biases introduced by the engine.

The dataset used in this study was curated and handscored for use in a wide-scale assessment program and includes 81 items from Mathematics (Math) and English Language Arts (ELA) from a national interim assessment program administered in grades 3–8 and 11. In this program, educators have historically been responsible for scoring these items. While a manual scoring approach grounds scoring in educator judgement, it also comes with a few weaknesses: (1) educators may not be trained in how to apply the scoring rubrics to responses; (2) approximately half of the responses are not scored by educators, meaning that the student does not receive a total test score and therefore information on her performance; and, (3) relatedly, educator scoring takes time away from instruction. Automated scoring was introduced into this program as a way to ensure the rubric is applied as intended by the program, to ensure all responses received a score, and to save educator time. The automated scoring engine described in this study produces a confidence measure that is used to flag poorly scored responses that educators are alerted to and can then score themselves; this approach ensures that educators have the opportunity to adjust scoring for such responses while not requiring them to score all responses.

Lastly, it is becoming increasingly important to evaluate fairness in deep learning systems. There have been notable approaches to fairness, one of which is Rater Scoring Modeling Tool (RMSTool) by ETS (Madnani & Loukina, 2016) which has been well established practices on fairness in automated scoring (Williamson et al., 2012). The approach we chose to evaluate bias used matched sampling so that subpopulations are compared with populations with the same score distribution. We believe that this approach corrects for student ability differences when subsampling, and hence, provides a more accurate indication of relative bias for particular subgroups (Fan, 2011).

In addition to the results, we describe the architecture of the engine and the underlying rationale for the architecture, describe handscoring and training/validation steps, and examine performance by selected key student groups. These results show that the engine meets typical industry criteria on engine performance for almost every item.

Automated Short Answer Scoring Engines

Automated scoring of unconstrained responses was introduced in 1968 by Ellis Page for essays (Page, 2003). Since that time several engines have been developed (eRater®, Knowledge Analysis Technologies™, CRASE®, IntelliMetric®, PEG) and have been used increasingly in both formative and summative assessment (Dikli, 2006). These engines have been shown to perform comparably to human raters (Shermis, 2013a), albeit not without criticism (Perelman, 2013). Current automated scoring engines are often combined with human scoring in some manner to produce scores on essays (Williamson et al., 2012). Many general automated scoring engines rely on a combination of techniques based on word frequency (Harris, 1954) in addition to expertly crafted features (Shermis, 2013b). Within the past few years, the automated scoring of essays has shifted from traditional recurrent and convolutional neural networks to transformer-based pretrained language models (Ormerod et al., 2021).

Automated scoring of short answer unconstrained responses is a relatively recent phenomenon. C-Rater (Leacock, 2003) was arguably the earliest engine that used Natural Language Processing methods with rule-based scoring to assess short answer items. Later models employed more statistically based methodology (Mohler, 2009). Unlike essays, automated short answer scoring has not historically matched human scoring, as evidenced by the Kaggle Automated Student Assessment Prize (ASAP) short answer competition (Shermis, 2015). Based on the results in that study, the authors argued that such engines are not ready for general operational use. Two other important datasets include the Powergrading dataset (Basu et al., 2013) and the SemEval 2013 Joint Student Response Analysis (SRA) dataset (Dzikovska et al., 2013). Since this time, strides have been made in automated scoring suggesting that these engines can be trained to produce results comparable to human scoring on at least a portion of items.

It is worth mentioning that there are many approaches that should work in short answer scoring; each comes with certain advantages and disadvantages. For example, a feature-based engine using regular expressions currently delivers state-of-the-art results for the Kaggle automated short-answer scoring dataset (Kumar et al., 2019) whereas features based on bag-of-word-based (e.g., Latent Semantic Analysis, or LSA) approaches often prove to be too brittle in the sense that they are often not generalizable to the use of language in a broader sample beyond the training sample.

More generally, frequency-based approaches ignore context in language (e.g., the addition of the word “not” before a word), which can be critical in determining whether a response is correct. Bag-of-word-based engines often provide baselines to compare the performance of other engines. For shorter responses, such as those found in the powergrading dataset, simple clustering methods deliver excellent results (Basu et al., 2013).Another approach related to clustering which has been proven to be effective involves comparing reference answers to student responses using tools like key words (Sakaguchi, 2015) and semantic similarity measures (Sultan, 2015).

Neural networks deliver state-of-the-art results in other fields such as speech-recognition, image classification (Szegedy, 2017), sentiment analysis (Zhilin Yang, 2019), and machine translation (Wu, 2016). We distinguish two types of approaches, traditional approaches such as those built with convolutional and recurrent layers with attention (Riordan et al., 2017) and pretrained language models fine-tuned for classification (Devlin, 2018). Firstly, neural networks consider patterns and/or sequence offering an approach in which rules are implicitly defined (or learned) and the context of a word is critically important. Secondly, pretrained transformer-based language models often require an order of magnitude more computational resources in training and inference. The state-of-the-art results for the SemEval-2013 student response task is currently given by a BERT model (Sung et al., 2019). The various ASAS datasets and corresponding approaches are summarized in Table 1. We refer to (Zhai, 2020) for a more comprehensive overview how the human–human agreements compare with performance of a range of automated scoring engines. Our intuition was that ensembling traditional neural network-based models with an LSA-based model would be efficient and would provide performance that would meet our requirements.

In an educational setting, the adoption of automated scoring system is a contentious issue (Anson, 2013). Page (2003) and Petersen (Page and Peterson, 1995) argue that the three criticisms of automated score are humanistic, defensive, and construct objections. At the heart of these criticisms is the fact that machines cannot understand human writing, measure writing differently than humans, and can be gamed. A fourth criticism is that the engines are “black boxes,” meaning that the way in which they produce scores is unknown to the user due to the use of vendors’ proprietary software and due to the complexity of these models.

One additional criticism of automated scoring, and machine learning in general, is that the field has not been sufficiently rigorous in evaluating bias. It is important that the equity of the resulting systems be analyzed (Murphy, 2019). An assessment of fairness for e-rater was considered (Bridgeman, 2009) where fairness was based on overall means and correlation statistics. More generally, as artificial intelligence becomes more commonplace in many aspects of our lives, we need to consider what bias may be present in training a neural network or machine learning algorithm more generally (Lee, 2018). We expect that a machine learning algorithm implicitly learns any biases present in the training set (Zou, 2018); however, by using the handscoring results as a control to keep student ability constant, we can discern whether the engine has introduced biases above and beyond those implicitly learned from the training set.

Table 1 The results of the Kaggle Automated Short Answer Scoring competition, Powergrading, and the dataset we present (writing, reading, and math) are in terms of QWK. The three way SemEval-2013 (SE13) tasks are in terms of weighted F1-score. Due to the different nature of the SemEval 2013, we have not provided length or number of prompts

	Prompts	# Train (avg)	Average Length (avg)	BaseLine (avg)	Best Known (avg)	Reference
Kaggle-ASAS	10	1363	48.4	0.653	0.791	(Kumar et al., 2019)
Powergrading	10	410	3.9	0.905	0.904	(Basu et al., 2013)
SE13 – Unseen Answer	197	4969	-	0.523	0.758	(Sung et al., 2019)
SE13 – Unseen Question	197	4969	-	0.520	0.648	(Sung et al., 2019)
SE13 – Unseen Domain	197	4969	-	0.554	0.612	(Sung et al., 2019)
Our Writing	49	3172.3	70.8	0.667	0.725	-
Our Reading	24	3249.7	40.9	0.663	0.735	-
Our Math	8	3169.9	56.3	0.698	0.756	-

It is in this context that we introduce an automated short answer scoring engine. As evidenced by the results, the engine performs well with respect to human raters on a wide variety of items in ELA and Math. In this paper, we describe the nature of the data used to train the engine, the scoring process, and rater agreement metrics pertaining to the quality of the data. We describe the methods and standards by which we trained and evaluated the performance of the engine. We also describe the engine and provide evidence of performance in the aggregate and for key student groups. We end the paper with a discussion of results as well as next steps.

Methods

Here, we describe the approach for engine training and validation. Critical elements to this approach are the items used in the study, the sample of responses, the way in which the items were handscored, the division of the sample into sub-samples for engine training and validation, the process for training the engine, the engine itself, and finally the metrics for evaluating engine performance.

Items

The items in this study came from a national interim assessment program for grades 3–8 and 11 in Math and ELA. The purpose of the assessment program is to provide educators with information that could be used to assess and monitor student knowledge and skills, with the goal of both improving student learning and revising instruction. The items were administered online with minimal constraints on what the students could type. With the exception of items that had dependencies in scoring and one item that had no data in which to train, all constructed response items available in the pool at the time were included in the study. In total, a set of 81 items were included in the study (Table 2). Items appeared across grades and two subject areas.

Each of the items were scored using a holistic rubric (Arter, 2000). All ELA items, both reading and brief writes, used a 2-point holistic rubric with score points of 0, 1, and 2. Six of the Math items were scored using 2-point rubrics with score points of 0, 1, and 2 while the remaining 2 items were scored with 3-point rubrics with score points of 0, 1, 2, and 3.

Typically, a 2-point holistic rubric for a short answer question might specify that a score of 2 points is achieved by answering a question with proper justification, 1 point might be assigned for an acceptable justification despite an incorrect answer or partial answer, while no points might be assigned if the criteria for 1 point have not been met. What constitutes a proper justification and a partially correct answer depends on what type of question being answered.

The ELA items were of two types: Brief Writes and Reading. In Brief Writes, students are asked to read a passage and write an introduction/beginning to, conclusion/ending of, or summary/explanation/extension of the passage or passage information. For example, given an article, we may ask the student what the main ideas

Table 2 Item counts for ELA and Math items by grade and rubric

Grade	ELA		Math	Total
	Holistic Rubric (0–1-2)		Holistic Rubric (0–1-2 or 0–1-2–3)	
	Brief Write Short Answer	Reading Short Answer	Short Answer	
3	6	2		8
4	7	2		9
5	6	3		9
6	6	4	2	12
7	8	3	1	12
8	7	5		12
11	9	5	5	19
Total	49	24	8	81

The ethnicities are labelled as follows: Unk is unknown, 0 is two or more races, 1 is Native American or Alaskan Native,2 is Filipino, 3 is Asian (non-Filipino), 4 is Hispanic or Latino, 5 is African American, and 6 is White

were being conveyed in the article where justifications are given by quoting particular sections of the passage supporting the summary. This engages a student's critical analysis. Another example includes providing an appropriate introduction or conclusion to an incomplete article, where justification relies on properly connecting ideas in the passage to the students writing, which engages a student's ability to synthesize text.

In Reading items, students are asked to read one or more passages and to answer a question using details or evidence from the passages. The passage may involve an article on a historic event, such as the Boston Tea Party, with key pieces of information, such as times, locations, or reasons for particular events. The student should be able provide any of the key pieces of information presented in the article with evidence for their choice. These types of questions are typically focused on testing a student's comprehension. Another example is when the student is presented with several sources on a topic, such solar energy, and the student is required to choose the which sources best support an argument with evidence from the sources to support their choice. A typical one-point response might say the main idea, but leave out a source, or give a detail that is not the main idea supported by the source.

Math items asked students to read stimulus material, answer a question about the material, and provide a justification or explanation for their answer. For example, a prompt may involve a two-dimensional graph with 3 different triangles. The student is asked which triangle has the greatest area. The intermediate calculations would be various side lengths, angles, of the triangles. These calculations would constitute an appropriate justification for the student's choice. A one-point response might state one of the areas incorrectly or get the areas correct with an incorrect conclusion. In this case a 3-point holistic rubric in mathematics may require a greater number of intermediate calculations to support that reasoning.

Data

Random samples of 4,000 constructed responses were drawn from the set of responses for each of the 81 items was obtained during operational testing in one Western state. These responses were randomly drawn from the prior year of administration. If 4,000 responses were not available within that year, the prior year was used in the sampling. This means that the total number of student responses obtained and labeled for training and validating the proposed engine was approximately 324,000.

The distribution of responses across key demographic variables in the full samples was examined. Because participation in the interim assessment program is voluntary by district, school, or educator, the population of interim testers cannot be expected to represent the state population. The purpose of the demographic analysis is to provide a description of key subgroups represented in the training and validation of the engine performance and to identify key subgroups with sufficient numbers to evaluate engine performance by key subgroup. The subgroups analyzed were: gender, ethnicity, Limited English Proficiency (LEP) status, economically disadvantaged (Title I) status, and disability (SPED) status. The contents for these variables were provided by the state during the rostering process.

The demographic results are displayed by item type (Table 3). The sample contained slightly more male than female students. Hispanic students were the dominant ethnicity (72% across all items), followed by White students (13.5%), Black students (5.3%), and Asian students (4.6%). Most students (66.8%) were Title I

Table 3 Percent distribution of sample, by subgroup and item type

Item Type		ELA-Brief Write	ELA-Reading	Math	Total
Number of Items		24	8	48	81
Gender	M	51.7	50.9	50.5	51.3
	F	48.3	49.1	49.5	48.7
Ethnicity	Unk	.08	.05	.03	.07
	0	1.9	2.2	1.9	2.0
	1	.5	.5	.5	.5
	2	1.7	1.9	1.8	1.7
	3	4.8	4.3	4.1	4.6
	4	72.4	71.2	72.0	72.0
	5	5.0	5.9	5.8	5.3
	6	13.4	13.6	13.6	13.5
	7	.3	.4	.3	.3
Title I	Y	65.7	67.6	70.9	66.8
	N	34.3	32.4	29.1	33.2
LEP	Y	23.9	22.4	14.8	22.5
	N	76.1	77.6	85.2	77.5
SPED	Y	5.8	6.1	7.0	6.0
	N	94.2	93.9	93.0	94.0

students. Across all items, LEP students made up 22.5% of the sample, although that percentage varied by item type (14.8% to 23.9%). Special education students made up 6% of the sample.

These data indicate that subgroup analysis may be appropriate for male/female students, for Hispanic/White/Black/Asian students, for Title I/non-Title I students, for LEP/non-LEP students, and for SPED/non-SPED students.

Handscoring

Responses were scored using standard procedures around handscorer recruitment, training, qualification, and monitoring outlined in the Standards for Educational and Psychological Testing (AERA, APA, & NCME, 2014). Rater training and qualification used the rangefinding materials developed by the assessment program. All raters possessed a bachelor's degree or higher. No specific teaching or subject-matter expertise was required as raters are trained to follow a prescriptive training and rubric-based model for scoring application. Raters were assigned to either math or ELA at the outset of training. Responses were routed to two independent readers, with any non-exact score routed to an expert third read. A backread rate of 3% was used to monitor raters. Validity papers were not used in monitoring readers, given the use of a 100% second read and expert resolution of non-exact scores. Raters were monitored using exact agreement rates with other raters and score point distributions.

The handscoring team also had the option to flag invalid responses for various reasons including that the response is blank, too short, duplicates the prompt text, off-topic, or the response was written in a language other than English. Any response flagged as invalid was routed to an expert reader for final scoring. The final, resolved scores were calculated as the expert read (if available) and the first human read otherwise. This process of admitting only valid responses have been automated in operational scoring.

Following handscoring, the quality of handscoring results was evaluated by examining the quadratic weighted kappa (QWK) and exact agreement/accuracy (Acc) between the two reads. Cohen's QWK statistic (Cohen, 1968) of two readers is given by

$$\kappa = 1 - \frac{\sum\sum w_{ij}x_{ij}}{\sum\sum w_{ij}m_{ij}}$$

where x_{ij} is the observed probability, $m_{ij} = x_{ij}\left(1 - x_{ij}\right)$, and

$$w_{ij} = 1 - \frac{(i-j)^2}{(k-1)^2}$$

given that k is the number of classes.

Williamson et al., (2012) recommend that the QWK for any item exceed 0.70. However, items whose QWK did not meet this threshold were still used in the modelling process because they were still in the item pool that was administered operationally.

Sub-Sampling

Once handscoring was complete, responses flagged as invalid were removed prior to model training and validation. The data were then divided into three sets: a training set consisting of 75% of the remaining data, a test set that consisted of 10% of the data, and a final validation set that consisted of 15% of the data. Sub-samples were stratified by score point to ensure that the distribution of score points was similar across the three samples. This is particularly important when score points are extremely unbalanced, as they were for many items in the sample.

The training set was used to train individual automated scoring models. The test set was used to evaluate model performance and to estimate parameters for statistical models that optimally combining model outputs. The validation set was used to validate the overall performance of the engine.

Engine Description

The short answer scoring engine presented consists of an ensemble of five different engines. Figure 1 presents a high-level architecture of the engine elements. The original response possessed various special characters and formatting tags from the system used by the students to input their responses. The procedure that removes all special characters and formatting tags is called cleaning and is simply an operational requirement of the system. This process is uniformly applied to both the training, test, validation, and in operation on live scoring samples. The cleaned responses are then submitted to five different engines, four of which are deep neural networks and one of which uses LSA (Deerwester et al., 1990). The neural networks each use the same architecture but differ in the choice of *word embedding*. The LSA model outputs a probability for each score point and the neural networks output a logit value for ordered groupings of score categories (0 vs. 1 and 2, 0 and 1 vs. 2, etc.). These

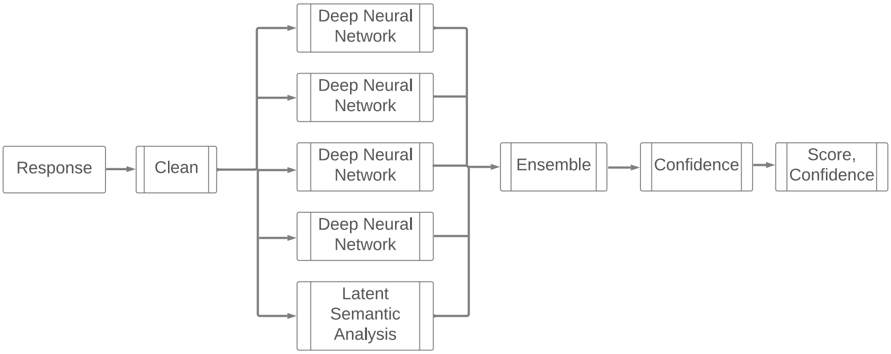

Fig. 1 Representation of the short answer automated scoring engine

outputs are then entered into an ensemble. The ensemble is the best-performing statistical classifier chosen from a set of eligible classifiers. A confidence level (in percentiles) is then provided alongside the predicted score.

Deep Neural Networks

The neural network processes are identical across the four neural networks. The difference between the neural networks is in the choice of word embeddings. A word embedding represents words in a high-dimensional vector space, known as a *semantic space*, in a manner that numerically represents the semantics or contextual use of the word (Turney, 2010). While the quality of these embeddings is known to improve with the size of the corpus used, there are other factors that seem to determine the appropriateness of a word embedding for a given task (Roberts, 2016). The nature of the embeddings depends on the nature of the corpora used to train the embedding. This is expected since the semantics of a word can depend on the context in which it is used.

Four word embeddings were used in the engine: two pretrained and publicly available embeddings and two constructed embeddings based on constructed responses. The two pretrained embeddings used corpora that are based on news corpora (Google[1]) and Wikipedia (GloVe[2]). Two constructed embeddings used responses the set of 81 items (item-specific) and a much larger set of response to a broad array of items beyond the 81 items (student-specific). The word embedding dimensions were built on the continuous bag of words method with dimensions that ranged from 100 for student-based embeddings and 300 for Google and GloVe embeddings.

One of the reasons for using four engines based on four different embeddings was that the desired architecture was required to serve as a single general architecture for items ranging from grade 3 to grade 11. By incorporating embeddings built on a range of corpora, we have a greater chance of at least one of the embeddings to better reflect the language used for each item. As long as at least one engine performs adequately, the ensembling procedure should be able to exploit those engines to perform sufficiently well. That is to say that by ensembling multiple engines, our system was much more robust to a variety of different item types and the language used by a variety of different grades. Furthermore, it is worth noting that the average performance of each individual engine falls short of the ensemble.

In this assessment program, the scoring rubrics do not evaluate spelling. Thus, if a student does misspell a word, the human rater is expected to evaluate the response, assuming that they understand the meaning of the misspelled word, and essentially ignore the misspelling when assigning a score. Given this, the automated scoring engine attempts to accurately correct mis-spelled words or handle

[1] https://colab.research.google.com/github/tensorflow/docs/blob/master/site/en/tutorials/text/word_embeddings.ipynb

[2] https://nlp.stanford.edu/projects/glove/

them robustly.[3] A variant of Peter Norvig's well-known spell-correction algorithm was used for spelling correction (Norvig, 2007a). This method only considers words that are not in a fixed vocabulary, hence, this method does not appropriately deal with real-word errors or grammatical errors.

Following spelling correction, each response word was mapped into the embedding space. Words that did not appear in the vocabulary were mapped to the space using the same value, corresponding to a zero vector. The order of words in the response was retained. Any response longer than 300 words was truncated, under the principle that it was likely most information in the response would occur in the first 300 words. Any response shorter than 300 words was padded with zeros. Once the responses were mapped to the embedding space, then they were entered into the neural network architecture. Note that in this setting, given a d-dimensional word embedding and that each response is artificially padded to 300 words, each response represented by a 300 × d-dimensional matrix.

Each neural network used two convolutional neural network layers, followed by a Gated Recurrent Units (GRU) network (Cho, 2012), followed by a series of fully connected feed forward networks. The total number of parameters estimated in the models varied from 177,249 (for 100-dimension embedding) to 241,249 (for 300-dimension embedding). Convolutional layers are thought to identify important sequences of words that are known to appear together, such as common phrases or common words that have been broken up by the spelling correction methods in a uniform manner. The recurrent network models the sequence of convolutional layer outputs to model the response as a whole. Finally, the feed forward layers served to reduce the dimensionality of GRU output, allowing for important dimension information to be retained and then aggregated. This means that each engine used both convolutional and recurrent layers, however, the dominant scoring mechanism is to be given by a set of recurrent layers. We note that even as an ensemble, these networks are an order of magnitude smaller than pretrained language models such as BERT whose base version consists of 110 million parameters (Devlin, 2018).

The networks are trained to mimic an order-dependent variant of the one-hot encoding: a score, x, between 0 and n is represented by precisely n binary values, specified by Boolean values $x \geq k$ for $1 \leq k \leq n$. In other words, for a rubric with scores 0, 1, and 2, two models are trained. One model predicts a score of 0, or score of 1 or 2. Another model predicts the score of 0 or 1, or a score of 2.

Latent Semantic Analysis

The LSA method relies on a classical machine learning framework that includes text preprocessing, feature extraction, and statistical modelling. Preprocessing methods include spelling correction, the conversion of words to their base word via stemming, and the removal of words that are infrequent in the training set vocabulary.

[3] The student and item-specific embeddings were that it was trained predominantly on a corpus in which there are many spelling mistakes. Approximately 90 thousand words in the 1.12 million-word vocabulary could be found in a large dictionary of correctly spelled words. Ad hoc inspection indicated that many incorrectly spelled words have a very high cosine similarity with their correctly spelled variants.

Note that words (after preprocessing) that do not appear in the final training set vocabulary are simply ignored by the later feature extraction and statistical modelling stages. In the context of the LSA, it is well known that words that appear sufficiently frequently in the training data can be associated with eigenvectors of words with similar semantics. This is known as the distributional hypothesis (Harris, 1954). When we remove this assumption, this association becomes unreliable, and hence, removing or even disregarding words that appear sufficiently infrequently guards against misinterpreted semantics.

After preprocessing, word occurrences for each response are counted and represented in a vector space of the same dimension as the size of the vocabulary used in training. While this matrix provides a lot of information, it does not associate words like “small” and “little.” To do this, a dimensionality-reduction analysis using LSA was employed using singular value decomposition, retaining the largest singular values. The expected result is that one can associate collection of words with similar meanings, based upon usage in the training sample. The number of singular values is made empirically based on model performance.

Support Vector Classification was used to model the LSA feature inputs to predict score. The radial basis kernel was used in the classification. The classifier outputs the probabilities of scores and provides the score with the highest probability. In addition to the LSA dimensions, different C values where modelled to examine the optimal degree of tolerance needed when making the classifications.

Ensembling is a process that inherently benefits from diversity. Ensembling models whose outputs are all very similar is not likely to yield any gains in accuracy. While we know that LSA generally produces baseline results that are lower than neural networks, we also know that the other engines were all modelled with similar processes, hence, the LSA was added to the ensemble to guard against model homogeneity for any particular item. The addition of LSA, especially for items with longer responses, yields accuracy gains without adding much in terms of the computational power required to score a response.

Ensembling

To determine a final score, the output of these models was the input space for a separate classifier that was chosen to be either a Logistic Regression, Random Forest, Adaptive Forest, or Gradient Boosting Classifier. The particular choice is different for each item and is based on results using a resampling procedure detailed in the training section. The multiclass versions of these classifiers are given by a family of “one-versus-all” classifiers. This classifier was fit to formthe output of the test set, which was then applied finally to the validation set to obtain our results.

Confidence

Finally, the engine was designed to produce an overall confidence that the engine has predicted a correct score. The confidence measure is calculated on the validation data using as inputs the output from each model and the ensemble. The dependent variable of the model is whether the prediction of the engine on a response is accurate (1) or inaccurate (0). Probit regression is used as the statistical method for production. The total number of inputs is $5n+2$, where n is the maximum score point. This is because there are n – 1 outputs from each of the four neural net models, n outputs from the LSA model, and n outputs from the ensemble classifier model. By using probit regression, we obtain an estimate of the probability curve that a given point in the parameter space is correct. This probability value is then considered the confidence value, or confidence that the score the engine predicted was accurate.

Finally, the confidence value was mapped to a confidence percentile value. This mapping is conducted using a separate, larger set of student responses that did not contain the data used to train and validate the engine.

Engine Training

All components of the engine were trained on the final adjudicated score, which was the best available score. For model training, the neural network and LSA methods differed. For neural networks, the responses were divided into batches of size 1,000 and were trained for 2,000 *epochs* or runs through the data. The model coming out of this process was considered final, even if it did not converge. In this case non-convergence is defined as a model predicting only one score point. Non-convergence was viewed as an ignorable issue because the ensemble would not weight value from that model. Competing LSA models were built with dimensions ranging from 5 to 200 alongside competing C values for the SVD (1 to 100) and the best-performing model, as measured by QWK, on the test set was considered final.

For ensembling, recall that four classifiers were examined. The final classifier was chosen using a resampling method. This method involved training each classifier on a subset of the test set and validating on the complement within the test set. This procedure is repeated and the classifier with the highest average QWK over different sub-samples on the test set was chosen. Once the classifier was identified, the classifier was re-fit on the full test sample. Once the final ensemble was refit, the validation data were scored.

Finally, it is important to note that the engine training process was conducted in parallel on three splits of the data. This means that the split into train/test/validation was conducted three separate times and the training process was conducted on each split. The purpose of this was twofold: (1) to be able to examine the sampling variability due to splits; and (2) to ensure that poor division of data into train/test/validation did not compromise the training process. The final split chosen for use in production was the test split that maximized the difference between the bootstrapped human–human QWK and the bootstrapped ensemble QWK on the above-mentioned test/train subdivision of the test set. This ensured that the validation set was blindly chosen since the validation statistics were only computed once the choice of split was chosen. That said, this overall process was unwieldly and did not produce results across items of value. Thus, the decision was made to use only one split in further trainings.

Evaluation Criteria

The performance of the engine on the validation sample was evaluated using QWK, Exact Agreement, and absolute Standardized Mean Difference (SMD). Absolute SMD is the absolute difference in the mean outcomes between the two scores normalized by dividing by the pooled standard deviations. For these indices, the engine-produced score was evaluated relative to the final resolved score. Then, this performance was evaluated relative to the two independent human scores.

Criteria commonly used in the industry to evaluate engine performance (Williamson et al., 2012; McGraw-Hill Education CTB, 2014; Pearson & ETS, 2015) were used in the evaluation of the engine. The criteria are as follows:

- The difference between the QWK of two humans and the QWK of the engine score and resolved score must be no greater than 0.10.
- The absolute SMD between the engine score and resolved score must be less than 0.15.
- The difference between Exact Agreement of two humans and Exact Agreement of engine score and resolved score must be no greater than 5.25%.

We should expect that the agreement of the engine with the best available score will exceed that of the two human raters, because the best available score should have less error than the "individual" rater scores. We should also expect that the agreement of the engine with each rater score should be comparable to that of the two raters.

While we indicate in this report when the engine does not meet these standards, the models are still used in production under the assumption that routing and teacher scoring of low-confidence responses will mitigate most issues, and that the benefit of providing automated scoring for all standalone items may outweigh the value of some items being entirely teacher-scored.

Subgroup Evaluation

The analysis on bias focused on comparing the mean scores of the engine relative to the human score and between groups on matched samples. Samples were matched on the final human resolved score. The purpose of the matching was to ensure that the evaluation interpretation was not impacted by the differences in examinee ability. We focused on mean scores over exact agreement or QWK because we were initially most interested in bias in score assignment rather than accuracy in score assignment. Finally, we focused on bias across items because we are primarily concerned with determining whether there is *engine bias*, rather than determining whether bias occurred for any given item. This evaluation of bias focused on the holistically scored items.

For bias evaluations, we define the target subgroup to be the subgroup of interest and the complement of the subgroup to be all subgroups not in the target subgroup. For instance, if the target subgroup is Black students, then the complement is the set of all ethnicities that are not Black in the sample.

We use the method of matched sampling to mitigate those differences in item difficulty and student ability (Rubin, 2006). As noted above, the indicator for "ability" in this situation is the final human score. In this setting, given a target subgroup of the population and an item, matched sampling first requires determining the maximal score distribution possible from the target subgroup and the complement of the target subgroup. This maximal score distribution is the minimum of the number of students from the subgroup and its complement that obtained a given score. After a score distribution is determined, we take a sample of student responses (with resampling) with that score distribution from both the subgroup and a complement. Note that for groups that represent a small percentage of the sample (e.g., LEP students), the maximal score distribution is the smaller student score distribution. For groups that represent a large percentage of the population (e.g., female students), the maximal score distribution is the minimal overlapping distribution between the target and the complement. Note that this method can change the target distribution; it is unclear how this impacts the results.

For each subgroup examination, 100 samples (without replacement) were drawn and the following statistics were computed: (1) average of the differences between the engine score and the final score for the targeted and the complement subgroup; and, 2) average of the differences between the engine score of the targeted subgroup and the engine score of the complement. The first two comparisons provide some indication of whether the engine-produced scores differ from the human scores and the second comparison provide some indication of whether the engine-produced scores differ between the target subgroup and the complement.

Results

In this section, we present the distribution of scores into train, test, and validation sub-samples, the performance of the engine relative to handscorers, the performance of the engine for responses above and below a 15th percentile confidence threshold, and the performance of the engine by student group.

Sub-Sampling

Table 4 presents the average number of invalid responses that were removed for each item type and grade, as well as the average size of the train, test, and validation samples. On average, there were about 2,800 responses used to train the models, about 372 to build the ensemble, and 560 to validate the final model. The average number of invalid responses varied by grade and item type but on average ranged from 88 to 454.

Table 4 Average number of invalid responses in Train, Test, and Validation Samples by Item Type and Grade

Grade	No. of Items	Invalid	Train	Test	Validation
ELA Brief Writes					
3	6	454.3	2,661.8	353.5	530.3
4	7	248.7	2,816.1	373.9	561.3
5	6	276.7	2,795.5	370.7	557.2
6	6	279.7	2,792.5	371.0	556.8
7	8	187.3	2,861.5	380.4	570.9
8	7	244.9	2,819.1	374.1	561.9
11	9	242.8	2,820.0	374.8	562.4
Total	49	269.4	2,800.4	371.9	558.3
ELA Reading					
3	2	258.0	2,809.5	373.0	559.5
4	2	152.5	2,888.5	383.0	576.0
5	3	133.3	2,902.3	385.3	579.0
6	4	133.3	2,902.8	385.5	578.5
7	3	187.3	2,862.3	380.0	570.3
8	5	167.8	2,876.4	382.4	573.4
11	5	225.6	2,833.8	376.0	564.6
Total	24	178.5	2,868.8	380.9	571.8
Math					
6	2	192.5	2,859.0	379.5	569.0
7	1	135.0	2,901.0	385.0	579.0
11	5	332.2	2,753.6	365.6	548.6
Total	8	272.6	2,798.4	371.5	557.5

Engine Performance

Engine performance on the ELA and Math short answer items is presented in this section. On average, the engine outperformed the human raters on the holistically scored ELA Brief Write and Reading items and performed similar to the human raters on the math items.

Table 5 presents the average exact agreements, QWK, and absolute standardized mean differences within each grade. Note that the human–human QWKs were on average less than 0.7 for most ELA items, but the engine-human QWKs were higher than 0.7 on average. For ELA Brief Write and Reading items, the engine tended to exhibit higher exact agreement (by 6–8%) and higher QWK (by about 0.1) and produce similar standardized mean scores. Additionally, the engine showed generally higher agreement with each rater than the raters did with each other, although the standardized mean differences of engine with the two rater scores are greater than that of the two raters. Unlike the ELA items, the engine showed generally lower agreement with each rater than the raters did with each other for the math items.

With regard to violations of the three rules outlined in the methods section, there was one SMD violation for ELA Brief Writes, zero violations for ELA

Table 5 Engine performance on holistic rubric items on the chosen held-out validation sets, by Grade and Item Type

Grade	No. of Items	Exact Agreement (%)				QWK				|SMD|			
		H1-H2	AS-HS	AS-H1	AS-H2	H1-H2	AS-HS	AS-H1	AS-H2	H1-H2	AS-HS	AS-H1	AS-H2
ELA Brief Write													
3	6	80.7	86.1	82.8	81.8	.714	.778	.720	.714	.029	.044	.031	.060
4	7	76.0	80.2	77.1	77.0	.645	.717	.660	.662	.030	.048	.072	.076
5	6	73.2	83.5	77.0	76.9	.618	.771	.667	.665	.038	.030	.074	.088
6	6	75.1	79.7	74.7	75.5	.630	.699	.618	.641	.021	.039	.094	.074
7	8	74.5	82.0	75.3	78.1	.672	.748	.665	.695	.050	.041	.096	.065
8	7	69.7	77.9	72.6	71.9	.567	.702	.625	.613	.024	.042	.057	.073
11	9	70.7	74.3	70.0	71.2	.626	.679	.626	.631	.042	.058	.099	.083
Total	49	74.0	80.2	75.2	75.8	.638	.725	.653	.659	.034	.044	.077	.074
ELA Reading Items													
3	2	83.1	86.2	84.4	84.2	.607	.678	.658	.620	.017	.048	.028	.044
4	2	79.9	84.2	81.4	80.4	.659	.667	.618	.605	.017	.043	.020	.037
5	3	76.8	81.3	77.2	77.8	.579	.637	.566	.569	.019	.022	.068	.060
6	4	77.3	88.0	80.7	81.0	.644	.813	.706	.711	.029	.026	.068	.083
7	3	76.3	82.7	74.9	78.2	.655	.760	.650	.688	.018	.016	.073	.056
8	5	73.6	80.8	74.1	75.9	.635	.734	.648	.674	.046	.045	.098	.106
11	5	74.0	80.7	75.8	75.3	.665	.765	.693	.689	.022	.026	.050	.051
Total	24	76.4	83.0	77.5	78.2	.638	.735	.655	.662	.026	.032	.064	.068
Math Items													
6	2	87.5	84.5	82.2	83.5	.861	.808	.779	.795	.025	.061	.098	.090
7	1	78.4	74.3	71.1	71.3	.783	.730	.694	.691	.026	.028	.009	.016
11	5	78.2	81.3	77.7	78.0	.703	.741	.678	.700	.028	.036	.056	.071
Total	8	80.6	81.2	78.0	78.5	.753	.756	.705	.722	.027	.041	.061	.069

H1-H2 is the human–human agreement. AS-HS is the agreement of the engine with the final resolved score. AS-H1 is the agreement of the engine with the first human score. AS-H2 is the agreement of the engine with the second human score

Reading items, and one Exact Agreement violation for Math items. Thus, across the 81 items, there were two violations (or 2.4%). This means that all QWK values were within 0.1 of the human–human agreements, one single SMD out of 81 was above 0.15, and one accuracy was lower than 10% of the human–human agreement.

On average, the engine performed slightly better than the human raters on the ELA Boolean binding items. Table 6 presents the average exact agreements, QWK, and absolute standardized mean differences within each grade. The engine showed similar exact agreement with each rater as the raters did with each other and lower QWK and higher absolute SMD with each rater as the rater did with each other. Not that the human–human QWKs were on average less than 0.7 for most ELA items, as were the engine-human QWKs; however, the engine-human QWKs trended closer to the 0.7 threshold.

Engine Performance by Confidence Level

As mentioned earlier, the automated scoring engine produces a confidence value that indicates how confident the engine is in the score it predicted. Responses with confidence percentile values less than 15 are flagged to educators for a scoring review. In Table 7,we present the Exact Agreement and QWK for records with confidence percentile values less than 15 (below threshold) and those 15 or above (above threshold). Note that the confidence model parameter was estimated on the validation set, so these statistics do not represent performance to be observed in an operational setting; rather, they illustrate more generally the likely pattern of performance above and below the threshold. This pattern suggests that responses with scores below the threshold are likely to be poorly scored by both the human and engine and should be routed for review. Responses above the threshold are likely to be better scored by both humans and the engine.

Engine Performance by Subgroup

In this section, we present the engine performance on groups matched on 'ability.' Recall that the matching approach used the final human score as the matching criterion. In this setting, given a target subgroup of the population and an item, a matched sample from the complement of the target subgroup was identified. This matching was conducted 100 times and the average of the results was computed.

The best-case scenario would be that average values are 0 across the items for the target and completement subgroups. Examining SMD values for comparing the engine to the human score for the target, most absolute values (averaged across items within an item type) were less than 0.05 and all were below 0.1. A few comparisons exceeded that threshold for the targeted subgroups: Asian students for Brief Writes, Hispanic students for all item types, and LEP and SPED students for Math. When examining SMD values for the matched complement, five of the subgroups values exceed the 0.05 threshold and for all item types (non-Asian, non-White, non-LEP, non-SPED, and Title I). Additionally, the magnitude of the SMD was greater

Table 6 The engine performance and human performance on holistic rubric-based items above and below 15th confidence percentile threshold

Grade	No. of Items	N		Exact Agreement (%)				QWK				|SMD|	
		Above	Below	Above Threshold		Below Threshold		Above Threshold		Below Threshold		Above	Below
				H1-H2	HS-AS	H1-H2	HS-AS	H1-H2	HS-AS	H1-H2	HS-AS		
ELA Brief Write													
3	6	432.8	76.8	83.8	90.3	63.0	62.6	.685	.771	.491	.382	.039	.184
4	7	462.4	82.1	79.9	85.5	54.5	50.4	.649	.712	.359	.284	.103	.286
5	6	457.8	82.0	76.3	88.6	55.9	55.3	.592	.788	.389	.315	.045	.191
6	6	450.8	82.2	77.3	84.1	63.1	55.6	.590	.674	.469	.340	.089	.295
7	8	472.5	84.6	76.4	86.5	64.2	56.9	.662	.776	.474	.275	.057	.136
8	7	461.3	82.9	72.9	82.2	52.1	53.9	.562	.693	.277	.253	.042	.174
11	9	464.8	83.7	73.1	78.0	57.6	54.0	.642	.708	.372	.248	.061	.251
Total	49	458.7	82.3	76.8	84.6	58.6	55.4	.627	.731	.402	.294	.063	.216
ELA Reading													
3	2	462.0	80.5	87.9	92.2	55.9	51.5	.481	.366	.360	.360	.159	.492
4	2	470.5	85.0	83.2	89.2	61.7	56.5	.586	.663	.524	.211	.060	.407
5	3	474.7	84.3	79.3	85.5	62.4	57.3	.476	.399	.486	.355	.167	.341
6	4	473.5	85.5	79.6	91.9	65.0	66.1	.636	.838	.350	.322	.035	.281
7	3	464.0	86.0	78.9	86.9	62.4	59.7	.642	.786	.405	.314	.025	.129
8	5	468.8	85.4	76.9	86.0	55.8	52.2	.633	.763	.405	.286	.085	.151
11	5	464.4	83.6	76.7	85.7	58.8	52.9	.671	.804	.444	.272	.045	.230
Total	24	468.4	84.5	79.3	87.8	60.1	56.5	.606	.700	.420	.301	.075	.260

Table 6 (continued)

Grade	No. of Items	N		Exact Agreement (%)				QWK				|SMD|	
		Above	Below	Above Threshold		Below Threshold		Above Threshold		Below Threshold		Above	Below
				H1-H2	HS-AS	H1-H2	HS-AS	H1-H2	HS-AS	H1-H2	HS-AS		
Math													
6	2	467.5	81.0	90.1	88.6	72.2	60.4	.861	.810	.698	.473	.085	.059
7	1	481.0	83.0	81.1	80.2	62.7	39.8	.797	.775	.652	.249	.034	.453
11	5	450.2	81.0	81.3	86.1	60.9	54.8	.675	.639	.450	.341	.120	.227
Total	8	458.4	81.3	83.5	86.0	64.0	54.3	.736	.699	.538	.363	.101	.213

H1-H2 is the human–human agreement. AS-HS is the agreement of the engine with the final resolved score. AS-H1 is the agreement of the engine with the first human score. AS-H2 is the agreement of the engine with the second human score.

Table 7 Means and standardized mean differences for matched samples for the human score and engine scores, as well as percentage of words correctly spelled and number of words in response, averaged across 100 matched samples

Target/Item Type	Avg. Sample Size	Means			SMD			Spelling		Word Count	
		HS	AS Target	AS Comp	AS-HS target	AS-HS comp	AS target -AS comp	Target	Comp	Target	Comp
Female	241.2	0.495	0.515	0.479	0.030	-0.027	0.057	0.846	0.845	65.1	55.9
Math	255.1	0.470	0.472	0.437	0.001	-0.048	0.050	0.844	0.841	62.2	50.6
Brief Write	238.7	0.555	0.574	0.540	0.025	-0.028	0.052	0.852	0.851	75.8	65.8
Reading	241.7	0.380	0.408	0.369	0.049	-0.019	0.067	0.834	0.834	44.3	37.5
Asian	25.5	0.784	0.776	0.704	-0.022	-0.122	0.099	0.851	0.845	76.9	71.9
Math	22.5	0.916	0.922	0.782	0.011	-0.165	0.175	0.834	0.832	74.9	67.2
Brief Write	27.2	0.797	0.771	0.725	-0.055	-0.118	0.063	0.861	0.853	86.7	81.9
Reading	22.9	0.714	0.737	0.634	0.035	-0.115	0.149	0.836	0.833	57.7	53.0
Hispanic or Latino	148.1	0.607	0.563	0.608	-0.068	0.000	-0.067	0.845	0.846	65.1	64.8
Math	145.3	0.613	0.549	0.597	-0.088	-0.023	-0.064	0.843	0.836	61.2	58.3
Brief Write	143.8	0.650	0.608	0.654	-0.067	0.002	-0.069	0.852	0.853	75.3	75.1
Reading	157.8	0.515	0.475	0.517	-0.063	0.003	-0.066	0.833	0.835	45.7	46.1
Black	27.5	0.415	0.427	0.436	0.018	0.037	-0.021	0.846	0.845	55.1	57.7
Math	29.6	0.346	0.343	0.364	-0.020	0.021	-0.044	0.842	0.845	48.2	53.2
Brief Write	25.1	0.467	0.497	0.489	0.047	0.039	0.007	0.853	0.851	65.0	67.4
Reading	31.8	0.334	0.312	0.353	-0.030	0.039	-0.070	0.834	0.834	37.2	39.5
White	72.3	0.627	0.614	0.588	-0.020	-0.060	0.040	0.842	0.846	63.8	66.4
Math	69.9	0.647	0.624	0.576	-0.030	-0.095	0.065	0.831	0.844	56.1	62.7
Brief Write	71.3	0.669	0.655	0.635	-0.023	-0.056	0.033	0.847	0.852	73.8	76.9
Reading	75.4	0.533	0.528	0.496	-0.010	-0.057	0.047	0.835	0.834	45.8	46.2
LEP	120.0	0.254	0.257	0.329	0.007	0.152	-0.146	0.842	0.847	48.4	51.2
Math	71.0	0.217	0.187	0.274	-0.088	0.107	-0.188	0.831	0.848	47.4	51.0

Table 7 (continued)

Target/Item Type	Avg. Sample Size	Means			SMD			Spelling		Word Count	
		HS	AS Target	AS Comp	AS-HS target	AS-HS comp	AS target -AS comp	Target	Comp	Target	Comp
Brief Write	125.6	0.290	0.292	0.374	0.006	0.164	-0.160	0.849	0.852	56.0	59.2
Reading	124.8	0.193	0.210	0.257	0.040	0.143	-0.103	0.831	0.835	33.2	35.1
SPED	30.9	0.228	0.217	0.300	-0.031	0.150	-0.176	0.827	0.846	42.3	50.1
Math	34.1	0.213	0.178	0.273	-0.083	0.125	-0.202	0.822	0.847	43.0	51.8
Brief Write	29.9	0.277	0.260	0.351	-0.045	0.139	-0.178	0.831	0.852	49.4	58.2
Reading	31.9	0.135	0.143	0.207	0.016	0.181	-0.163	0.819	0.835	27.7	32.9
Non-Title I	178.7	0.598	0.591	0.561	-0.012	-0.057	0.045	0.846	0.845	64.0	64.9
Math	153.1	0.620	0.608	0.551	-0.016	-0.091	0.074	0.837	0.842	61.4	60.0
Brief Write	183.7	0.636	0.631	0.605	-0.012	-0.052	0.040	0.853	0.851	73.4	75.2
Reading	177.0	0.511	0.504	0.476	-0.011	-0.055	0.044	0.835	0.833	45.8	45.6

HS = final human response score for matched samples. Target = the focus on the investigation, as indicated by the category of subgroup (e.g., Females). Comp = the sample generated from the non-target sample (e.g., non-Females). Spelling = percentage of words matched to a dictionary than 0.1 for non-Asian students, non-LEP students, and non-SPED students. The non-Asian matched sample was assigned lower scores by the engine than by human score4s, and the opposite was true for the non-LEP and non-SPED students.

Examining the SMD of the engine scores for the two groups (target vs. complement), almost all SMDs exceeded 0.05 in magnitude, except for Black student comparisons (all item types), White student comparisons (Brief Writes and Reading items), and non-Title I student comparisons (Brief Writes and Reading items). SMD differences above 0.1 in magnitude occurred for Asian student comparisons (Math, Reading), and LEP and SPED student comparisons (all items).

For interpretive purposes, we provide the proportion of correctly spelled words (compared to a dictionary) in the responses for each group, as well as the average length of responses (in words) for each target group. The spelling results indicate that on average, the matched groups had similar levels of correctly spelled words. We do see differences greater than 0.1 between SPED and non-SPED students for all item types, and LEP and non-LEP students in Math, as well as White and non-White students in Math. The length of the responses, as measured by word count, varied by group for the matched samples. Note, however, that the difference was not generally associated with the SMD magnitude. For instance, females tended to write longer responses than the matched non-females (here, males), but overall the SMD differences for females and their complement were small relative to the human score and for engine scores relative to one another. For the Asian student comparisons, the difference in lengths was smaller but the magnitude of SMD differences were larger. These results suggest that there are other factors—presumably the content of the response—that are more strongly associated with score.

We also performed matched sampling at an individual engine level to see whether there was more or less bias introduced by any one of the engines or by the ensembling procedure itself. The simplest way to express the bias for each engine is to examine z-scores using match score distributions. Given a subpopulation, we can calculate the z-score by estimating the difference between the average machine scores for that subpopulation and the expected machine scores for any subpopulation with the same human assigned score distribution normalized by the standard deviation. By taking students with the same human assigned score distribution we mitigates any effects caused by differences in student abilities within each item (Fan, 2011). Overall we see the most absolute bias from the ensembling procedure itself, which we did not expect, followed by the BoW model.

In the case of reading and writing for LEP students, this bias seems to be driven by the Google and BOW engines while the ensembling procedure seems to be introducing negative bias overall. We suspect that some of this bias is based on real-word spelling errors, that are more prevalent in LEP than non-LEP students. While spelling correction may be mitigating some of this bias, the glove, student and essay embeddings are built from corpora with less formal language which might be more robust to real-word errors. If a word is not recognized or used incorrectly, the uncertainty causes the engine to regress to the most common scores in the distribution, which are almost uniformly lower than they are higher. Context aware spelling and grammar correction might do a better job at mitigating this bias. As for some other strong bias, such as male/female bias, we suspect some of this is driven by average length discrepancies. Typically, subpopulations that write more on average are simply more likely to fulfill the criteria of the rubric (Table 8).

Lastly, we investigated whether this bias might be mitigated by the confidence mechanism. The Spearman correlation coefficients between the average ensemble's

Table 8 A summary of our estimates for the bias in each engine as defined by the z-score for matched samples. This is the standardized difference between average model scores for each subpopulation and the expected model scores for samples of students with the same human score distribution

Target/Item Type	Avg. Sample Size		z = (Target – Expected)/STD					
		HS avg	Ensemble	Google	GloVe	Student	Essay	BoW
Female	241.2	0.495	0.95	0.48	0.63	0.84	0.77	1.01
Math	255.1	0.47	1.05	0.40	0.24	0.73	0.78	0.74
Brief Write	238.7	0.555	0.83	0.20	0.78	0.88	0.71	1.00
Reading	241.7	0.38	1.20	1.08	0.60	0.92	0.99	1.07
Asian	25.5	0.784	0.65	0.80	0.54	0.61	0.60	0.49
Math	22.5	0.916	1.30	1.58	1.20	1.20	1.25	0.81
Brief Write	27.2	0.797	0.32	0.46	0.41	0.40	0.39	0.10
Reading	22.9	0.714	1.08	1.29	0.51	0.88	0.92	1.08
Hispanic or Latino	148.1	0.607	-0.62	-0.29	-0.55	-0.42	-0.37	-0.41
Math	145.3	0.613	-1.03	-0.84	-1.14	-1.36	-0.97	-0.83
Brief Write	143.8	0.65	-0.57	-0.02	-0.37	-0.26	-0.35	-0.28
Reading	157.8	0.515	-0.62	-0.78	-0.73	-0.51	-0.25	-0.52
Black	27.5	0.415	-0.13	-0.19	-0.14	-0.05	-0.23	-0.12
Math	29.6	0.346	-0.37	-0.95	-0.46	-0.44	-0.31	-0.33
Brief Write	25.1	0.467	0.05	-0.12	-0.03	0.06	-0.15	0.00
Reading	31.8	0.334	-0.52	-0.17	-0.36	-0.19	-0.37	-0.22
White	72.3	0.627	0.38	0.22	0.43	0.33	0.36	0.31
Math	69.9	0.647	0.66	0.50	0.82	0.89	0.58	0.50
Brief Write	71.3	0.669	0.33	0.29	0.31	0.38	0.23	0.26
Reading	75.4	0.533	0.44	0.02	0.59	0.14	0.61	0.33
LEP	120	0.254	-0.94	-0.83	-0.67	-0.67	-0.89	-0.99
Math	71	0.217	-1.27	-0.51	-1.02	-0.75	-1.45	-1.23
Brief Write	125.6	0.29	-1.02	-0.98	-0.52	-0.78	-0.86	-0.95
Reading	124.8	0.193	-0.59	-0.78	-0.71	-0.44	-0.73	-0.90
SPED	30.9	0.228	-0.72	-0.65	-0.29	-0.39	-0.67	-0.61
Math	34.1	0.213	-0.64	-1.99	-0.50	-0.47	-0.47	-0.38
Brief Write	29.9	0.277	-0.78	-0.71	-0.53	-0.40	-0.71	-0.65
Reading	31.9	0.135	-0.65	-0.29	0.18	-0.42	-0.71	-0.61
Non-Title I	178.7	0.598	0.50	0.31	0.43	0.44	0.37	0.50
Math	153.1	0.62	0.87	0.81	1.11	0.80	0.87	0.98
Brief Write	183.7	0.636	0.60	0.31	0.30	0.58	0.33	0.50
Reading	177	0.511	0.18	0.30	0.31	0.13	0.26	0.37

confidence and the Z-scores for subpopulation bias were -0.736 indicating that the ensemble on average seems to be less confident by those affected positively by bias than those affected negatively by bias. This may be due to the inherent confidence in low scores compared to high scores because the engines have usually been trained on many more examples of low scores than high scores. This might be better mitigated by using better weighting mechanisms in the training procedure for the confidence models and in the ensembling and engine training. This set of results suggest that further investigation into differences between the Asian/non-Asian, LEP/non-LEP, and SPED/non-SPED is warranted.

Discussion

These results suggest that the use of preprocessing, deep learning using neural networks, and ensembles of deep learning and LSA-based models can produce better than human-quality results in ELA and similar results to human scores in math. In terms of aggregate performance, 79 items out of the 81 met the performance criteria. The use of the confidence metrics indicates that low confidence responses do tend to be poorly scored by humans and the engine, and the responses above the threshold tend to be adequately scored by both. In terms of bias, the engine appears to match human scoring well (in the aggregate) for each subgroup. However, when we compare engine performance of target subgroups to their complement, some issues arise for Asian, LEP, and SPED students. These differences merit further study.

In terms of use by educators, these models were deployed for teacher use in 2018–2020 and we have received no questions from educators about the quality of scoring. This may be due either to educators' agreement with the scores or to educators not examining the scores relative to student writing. Our analysis indicates that very few educators change engine scores (about 1% of scores have been changed); however, we do not know how many responses and scores were reviewed by the educators. A follow-up study to validate scores in the field would be of value.

This study had two key limitations. First, the use of multiple splits during the training phase could have influenced the final results. While care was taken to ensure that the split choice did not involve the validation sample, the use of multiple splits is atypical. Future calibrations with the engine have used only one split, with similar results. Second, the evaluation of engine performance by confidence threshold is somewhat contaminated as the validation sample was used to estimate confidence weights. A better approach would be to examine performance on a different sample.

Aside from future validation activities, three next steps are planned. The first step is to examine further the bias toward or against targeted subgroups (students with disabilities, non-native English speakers, etc.) to understand the nature of the bias and to correct for it. The second step is to examine whether the complexity of the model can be reduced to limit calibration time and model deployment requirements while also retaining scoring quality. This work is already underway and is yielding good success. The third step to explore the notion of explainability of these models. Work in explainability has primarily been conducted in vision but there are papers that provide some direction and value in this area.

Conclusion

We showed that an ensemble of several recurrent neural network engines and an LSA approach gives a performance that is above that of humans in the above described manner. In doing so we have presented a manner by which we can deliver fast, low-cost, and statistically accurate feedback for an interim assessment program. While we firmly believe that the bar for delivering an analogous solution to summative assessment should be much higher, we believe that more current methods will be able to achieve these results soon.

In the process of designing, developing, implementing, and testing this engine, several key developments have been made in the domain of NLP. Many current state-of-the-art results have been made recently using general language models built on large bodies of text (Vasawni, 2017; Zhilin Yang, 2019). These models have already proven to be effective in scoring essays (Ormerod et al., 2021). We expect that the discrepancy between the way in which language is used in the training and application may pose some difficulty analogous to the problems faced with word embeddings; however, pretrained models may prove to be an excellent starting point for training models based on student constructed responses.

We end with brief comments about perceptions that computers can never be sophisticated enough to score student writing. The ability of models to generate text that is difficult for humans to detect as artificially generated suggests that the ability to better model human writing – and thereby score and provide informative feedback – is on the near horizon. That said, there are a number of issues to address and these are targeted toward Ellis Page's three criticisms and the fairness issue: (1) computers currently do not understand language in the way humans do; (2) understanding how scores are arrived at (the explainability problem) is an important missing element in the deep learning work; (3) ensuring that engines are robust to cheating behavior; and (4) ensuring that scores produced by the engine are fair to all examinees.

Data Availability The data provided may contain personally identifiable information and is not publicly available.

Code Availability The code used to model these responses is not generally available.

Declarations

Conflicts of Interest/Competing Interests Not Applicable.

References

American Educational Research Association, American Psychological Association, National Council on Measurement in Education. (2014). *Standards for educational and psychological testing*. American Educational Research Association.

Anscombe, F. J. (1948). The transformation of Poisson, binomial and negative-binomial data. *Biometrika*, 246–254.

Anson, C. S. (2013). *NCTE position statement on machine scoring: Machine scoring fails the test.*

Arter, J. (2000). Rubrics, scoring guides, and performance criteria: Classroom tools for. *The Annual Conference of the American Educational Research Association.* New Orleans.

Basu, S., Jacobs, C., & Vanderwende, L. (2013). Powergrading: A clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics*, 391–402.

Bridgeman, B. C. (2009). Considering fairness and validity in evaluating automated scoring. *National Council on Measurement in Education.* San Diego.

Cho, K. v. (2012). Learning phrase representations using rnn encoderdecoder for statistical machine translation. *preprint arxivs*, 1406.1078.

Cohen, J. (1968). Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 213.

Darling-Hammond, L. J. (2013). *Criteria for high-quality assessment*. Stanford Center for Opportunity Policy in Education.

Deerwester, S., Dumais, S., Furnas, G., Landauer, T., & Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American Society for Information Science, 41*(6), 391–407.

Devlin, J. M.-W. (2018). *BERT: Pre-training of deep bidirectional transformers for language understanding*. arXiv preprint arXiv:1810.04805.

Dikli, S. (2006). An overview of automated scoring of essays. *The Journal of Technology, Learning and Assessment*.

Dzikovska, M. O., Nielsen, R. D., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., & Dang, H. T. (2013). *Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge.* NORTH TEXAS STATE UNIV DENTON.

Esteva, A. B. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 115.

Fan, Xitao, & Nowell, Dana L. (2011). Using propensity score matching in educational research. *Gifted Child Quarterly, 55*(1), 74–79.

Gewertz, C. (2013, June 9). States Ponder Costs of Common Tests. *Education Week*, pp. 20–22.

Gong, T. a. (2019). An Attention-based Deep Model for Automatic Short Answer Score. *International Journal of Computer Science and Software Engineering*, 127–132.

Hand-Scoring Rules. (2016). Retrieved from http://www.smarterapp.org/documents/Smarter_Balanced_Hand_Scoring_Rules.pdf. Accessed 18 June.

Harris, Z. S. (1954). Distributional structure. *Word*, 146–162.

Hochreiter, S. a. (1997). Long short-term memory. *Neural computation*, 1735–1780.

Kumar, Y., Swati A., Debanjan M., Rajiv R. S., Ponnurangam K., & Roger Z. (2019). Get it scored using autosas—an automated system for scoring short answers. In Proceedings of the AAAI Conference on Artificial Intelligence, *33*(1), pp. 9662–9669.

Leacock, C. a. (2003). C-rater: Automated scoring of short-answer questions. *Computers and the Humanities*, 389–405.

Lee, N. T. (2018). Detecting racial bias in algorithms and machine learning. *Journal of Information, Communication and Ethics in Society*.

Madnani, N. & Loukina, A. (2016). RSMTool: A Collection of Tools for Building and Evaluating Automated Scoring Models. *Journal of Open Source Software*.

McCurry, D. (2010). Can machine scoring deal with broad and open writing. *Assessing Writing*, 118–129.

McGraw-Hill Education, C. T. (2014). *Smarter balanced assessment consortium field test: Automated scoring research studies (in accordance with smarter balanced RFP 17).*

Mikolov, T. I. (2013). Distributed representations of words and phrases and their compositionality. . *Advances in neural information processing systems*, 3111–3119.

Mohler, M. a. (2009). Text-to-text semantic similarity for automatic short answer grading. *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics.* (pp. 567–575). Association for Computational Linguistics.

Murphy, R. F. (2019). *Artificial Intelligence Applications to Support K–12 Teachers and Teaching*. RAND Corporation.

Norvig, P. (2007a). Retrieved from How to write a spelling corrector: http://norvig.com/spell-correct.html. Accessed July 2018

Norvig, P. (2007b). *How to write a spelling corrector*. Retrieved from How to write a spelling corrector: http://norvig.com/spell-correct.html. Accessed July 2018

Ormerod, C. M. & Harris, A. E. (2018). Neural network approach to classifying alarming student responses to online assessment. *arXiv preprint*, 1809.08899.

Ormerod, C. M., Malhotra, A., & Jafari, A. (2021). Automated essay scoring using efficient transformer-based language models. *arXiv preprint arXiv:2102.13136*.

Osgood, C. E. (1964). Semantic differential technique in the comparative study of cultures 1. *American Anthropologist*, 171–200.

Page, E. B. & Petersen, N. S. (1995). The computer moves into essay grading: Updating the ancient test. *Phi delta kappan*, 561.

Page, E. B. (2003). Project Essay Grade: PEG. In M. D. Shermis (Ed.), *Automated essay scoring: A cross-disciplinary perspective* (p. 43). New Jersey: Lawrence Erlbaum Associates.

Pearson and ETS. (2015). Research results of PARCC automated scoring proof of concept study. Retrieved from http://www.parcconline.org/images/Resources/Educator-resources/PARCC_AI_Research_Report.pdf. Accessed Sept 2019.

Perelman, L. C. (2013). Critique of Mark D. Shermis & Ben Hammer, Contrasting state-of-the-art automated scoring of essays: Analysis. *Journal of Writing Assessment*.

Powers, D. E. (2002). Stumping e-rater: challenging the validity of automated essay scoring. *Computers in Human Behavior*, 103–134.

Rajpurkar, P. J. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:*, 1606.05250.

Riordan, B., Horbach, A., Cahill, A., Zesch, T., & Lee, C. (2017). Investigating neural architectures for short answer scoring. In Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications, pp. 159–168.

Roberts, K. (2016). Assessing the corpus size vs. similarity trade-off for word embeddings in clinical NLP. *Proceedings of the Clinical Natural Language Processing Workshop*, (pp. 54–63). Osaka, Japan.

Rubin, D. B. (2006). *Matched sampling for causal effects*. Cambridge University Press.

Sakaguchi, K. M. (2015). Effective feature integration for automated short answer scoring. Proceedings of the 2015 conference of the North American Chapter of the association for computational linguistics: Human language technologies, (pp. 1049–1054).

Sato, E. R. (2011). *SMARTER balanced assessment consortium common core state standards analysis: Eligible content for the summative assessment. Final Report.* Smarter Balanced Assessment Consortium.

Shermis, M. D. (2013a). *Contrasting state-of-the-art automated scoring of essays: Analysis.* Annual national council on measurement in education meeting.

Shermis, M. D. (2013). *Handbook of automated essay evaluation: Current applications and new directions*. Routledge.

Shermis, M. D. (2015). Contrasting state-of-the-art in the machine scoring of short-form constructed responses. *Educational Assessment*, 46–65.

Silver, D. A. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 484.

Smith, C. (2017). *iOS 10: Siri now works in third-party apps, comes with extra AI features.* BGR.

Sultan, M. A. (2015). Fast and easy short answer grading with high accuracy. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (pp. 1070–1075).

Sung, Chul, Tejas Indulal Dhamecha, and Nirmal Mukhi. (2019). Improving short answer grading using transformer-based pre-training. *International Conference on Artificial Intelligence in Education*, (pp. 469–481).

Szegedy, C. S. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. *In Thirty-First AAAI Conference on Artificial Intelligence.*

Tomas Mikolov, K. C. (2013). Efficient Estimation of Word Representations in Vector Space. *Proceedings of Workshop at ICLR.*

Turney, P. D. (2010). From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 141–188.

Vaswani, A. N. (2017). Attention is all you need. *Advances in neural information processing systems.*, 5998–6008.

Vogels, W. (2017). *Bringing the Magic of Amazon AI and Alexa to Apps on AWS*. Retrieved from All Things Distributed: www.allthingsdistributed.com

Williamson, D. M., Xiaoming X., & Breyer, F. J. (2012). A framework for evaluation and use of automated scoring. *Educational measurement: issues and practice*, 31(1), 2–13.

Wu, Y. M. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation. *arxiv preprints*, 1609.08144.

Zhilin Yang, Z. D. (2019). XLNet: Generalized Autoregressive Pretraining for Language Understanding. *preprint Paper arxiv*, 1906.08237.

Zhou, Z.-H. J. (2002a). Ensembling neural networks: many could be better than all. *Artificial Intelligence*, 239–263.

Zhou, Z.-H. J. (2002b). Ensembling neural networks: many could be better than all. *Artificial Intelligence*, 239–263.

Zhai, X. Y. (2020). Applying machine learning in science assessment: a systematic review. *Studies in Science Education*, 111–151.

Zou, J. a. (2018). AI can be sexist and racist—it's time to make it fair. *Nature*, 324–326.