Enhancing Auto-scoring of Student Open Responses in the Presence of Mathematical Terms and Expressions

Authors Blinded for Review

Blinded for Review

Abstract. With the greater application of machine learning models in educational contexts, it is important to understand where such methods perform well as well as how they may be improved. As such, it is important to identify the factors that contribute to prediction error in order to develop targeted methods to enhance model accuracy and mitigate risks of algorithmic bias and unfairness. Prior works have led to the development and application of automated assessment methods that leverage machine learning and natural language processing. The performance of these methods have often been reported as being positive, but other prior works have identified aspects on which they may be improved. Particularly in the context of mathematics, the presence of non-linguistic characters and expressions have been identified to contribute to observed model error. In this paper, we build upon this prior work by observing a developed automated assessment model for open-response questions in mathematics. We develop a new approach which we call the "Math Term Frequency" (MTF) model to address this issue caused by the presence of non-linguistic terms and ensemble it with the previously-developed assessment model. We observe that the inclusion of this approach notably improves model performance. Finally, we observe how well this ensembled method extrapolates to student responses in the context of Algorithms, a domain similarly characterized by a large number of non-linguistic terms and expressions. This work represents an example of practice of how error analyses can be leveraged to address model limitations.

Keywords: Online learning platforms · Math-terms · Open-ended responses · Automated assessment · Machine Learning · Natural Language Processing · Mathematics.

1 Introduction

Advancements in artificial intelligence and machine learning research have led to greater integration of prediction models into educational contexts through computer-based learning systems. Often emerging from learning theory or for the purpose of addressing an identified problem of practice, machine learning models are being used to direct teacher attention to students in need [16, 8], aid

in assessment [11, 32, 23, 6, 1], and track student learning over time [10, 14, 20, 24]. As these methods and models of student learning become deeply integrated into normal instructional and educational practices, it becomes increasingly important to understand the strengths and weaknesses in their application. Within this, it is important to not only identify areas where existing student models underperform, but similarly important to develop targeted methods to improve such models to mitigate risks to fairness.

Several prior works have leveraged machine learning methods to automate assessment for student work [21, 30]; many of these methods have emerged to help teachers save time in providing feedback to students and allow them to focus their attention on helping students who are in most need of aid. It is not surprising, given the ubiquitous challenges posed by assessment across educational contexts, that automated assessment methods have been proposed, developed, and applied in a range of domains. They commonly address one of two types of problems: close-ended and open-ended. In the case of close-ended problems, where there is a finite number of accepted correct answers, auto-scoring methods can apply simple matching techniques to compare the student answer with the list of correct answers and consistently achieve near-perfect accuracy. In regard to open-ended problems, however, the correctness of student responses is more subjective, where teachers commonly assess students based on an explicit or implicit rubric that identifies key points that must be included in a student response to sufficiently demonstrate comprehension. Due to the numerous challenges that this poses to automated assessment, existing methods commonly apply natural language processing (NLP) to build a high-dimensional representation of student responses that is then combined with various machine learning approaches (e.g. [28, 33, 9, 13]).

In consideration of the challenges in assessing open-ended problems, mathematics-based domains make developing automated assessment models even more difficult. In such domains, including mathematics, statistics, physics, chemistry, and even computer science, student responses often exhibit a combination of natural language and various non-linguistic terms such as numbers, mathematical expressions and operators; this makes automatic assessment more difficult as most traditional NLP techniques were not designed for such a context, with a few recent exceptions [1, 18, 25, 29]. Recent work has identified that the existence of non-linguistic terms is positively correlated with model prediction error in models that have outperformed existing benchmarks in this context [2].

Students who are being assessed by these automated methods in practice are in danger of being unfairly penalized due to the number of non-linguistic terms they use in their responses. While systems have attempted to mitigate this risk by involving the teacher in the assessment process (e.g. [1]), this work attempts to do better by developing a simple, targeted method to resolve this problem. Drawing inspiration from closed-ended assessment methods, we call this proposed method the "Math Term Frequency" (MTF) model and demonstrate how it can be combined with previously-developed assessment models to improve performance. Specifically, this work addresses the following research questions:

1. How does accounting for non-linguistic terms through our MTF model affect the performance of auto-assessment methods on existing benchmarks?
2. Does our MTF method reduce the correlation between non-lingustic terms and model prediction error?
3. How well does our MTF method extrapolate to other domains where student responses contain non-linguistic terms?

2 Background

As briefly introduced in the previous section, there are many prior works that have focused on automating the assessment (i.e. grading or scoring) of student answers to open-ended questions. Much of this work has leveraged varying approaches that leverage NLP and machine learning. Methods such as C-rater [19] uses techniques to normalize student responses that vary across syntactic, morphological structure, pronouns, and synonyms to estimate the correctness of student responses to open-ended questions. Other approaches have explored the use of clustering approaches to grade student responses [4, 5]. Recent approaches have used deep learning methods that use high-dimensional representations of student work and compare them to exemplar samples, such as in [28] and [33].

While not universally the case, a majority of recent works in NLP have leveraged or expanded upon this idea of creating high-dimensional representations, referred to as embeddings, of student answers (e.g. Word2Vec [22] and GloVe [26]). However, word embeddings can often lose information about the context of words within the sentence, leading to developments in sentence embedding methods such as Universal Sentence Encoders [7] and SBERT [27]; the later of these was built from Bidirectional Encoder Representations from Transformers (BERT; [12]).

This work observes a recent automatic assessment method that draws from many these methods and concepts [2]. This method, referred to as the SBERT-Canberra model, utilizes a similarity-matching approach using pre-trained SBERT embeddings. Outperforming previous benchmarks in predicting teacher-provided scores for student answers to mathematics open-response problems, this method works by identifying "similar" student answers using a measure of Canberra distance [17] between embeddings; predicted scores are then produced by taking the given score for the most-similar student answer from a pool of historic responses.

In that prior work, an error analysis was conducted, identifying three key takeaways: 1) problem-level factors explained the majority of variance of prediction error (beyond that of answer- and teacher-level factors), 2) the presence of images in student answers had the highest correlation with prediction error among answer-level factors, and 3) the presence of non-linguistic terms (i.e. numbers, equations, and mathematic expressions) exhibited the second-highest correlation with prediction error among answer-level factors. This work explores the first and third points to develop simple approaches to target and mitigate these weaknesses.

3 Methodology

3.1 Dataset

To explore and examine the methods proposed in this work, we observe two datasets consisting of student answers to mathematics open-response questions. These datasets were collected from the BLINDED SYSTEM [3] and contains 150,477 student responses from 27,199 students for 2,076 open-ended math problems scored by 970 unique teachers (where each response was scored by a single teacher); this dataset is the same used to establish benchmark results [1] and is used to directly compare performance against models presented in prior work[1, 2]. Teachers scored responses based on a 5-point integer scale ranging from 0 to 4 , with a 4 indicating a very strong and a 0 indicating a very weak response. In this dataset, all the empty student responses and responses containing only images are omitted when training and evaluating the model.

The second dataset used in this paper was similarly used in prior work to conduct an error analysis to identify factors that correlate with prediction error [2]. This dataset is similarly comprised of student open responses collected using the BLINDED TOOL and contains 30,371 scored student responses from 1,628 students for 915 unique open-response questions assessed by 12 different teachers. In addition to the student answers and teacher-provided scores, this dataset was expanded in that prior work to include other measures describing student answers including the length of the response, the average length of words (in characters) in the response, a count of numeric terms, a count of mathematical operators, the percent of the response (the proportion of words) containing non-linguistic terms[1], and whether the answer contains an image (e.g. usually a picture taken of student work and uploaded as part of their response).

3.2 The SBERT-MTF Model

The methods presented in this work target the specific problem of non-linguistic terms contributing to prediction error. The previously-developed SBERT-Canberra model outperformed previous decision-tree- and deep-learning-based approaches [1] by leveraging pre-trained Sentence-BERT embeddings. The use of pre-trained embeddings provides several advantages in that they are commonly built using very large corpuses of data; by training such embedding methods on such sources as Wikipedia or online news sources, the models can learn the semantic meaning of words and sentences based on their proximity to other words and sentences within observed documents. In short, pre-trained models can learn language representation from large datasets that can then be used to increase the predictive power in smaller datasets. The challenge, however, is that only a finite number

[1] This feature is named as "equation percent" in the prior work and referred to in this paper with the same name for consistency, though we clarify that it is a proportion of all non-lingustic terms rather than being limited to just equations as the name suggests.

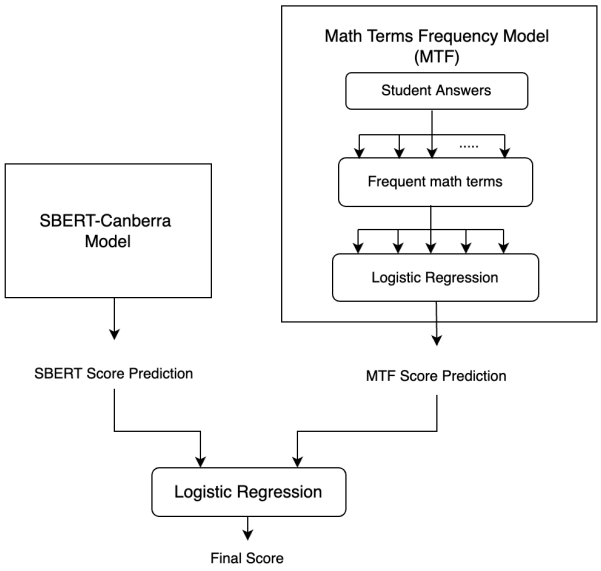

Fig. 1. The design of the ensembled SBERT-MTF method, that suggests scores for student open responses.

of words (and sentences, by extension) can be recognized by these methods; traditionally, unrecognized words and phrases may be given a default embedding. When observing non-lingustic terms such as numbers and expressions, many such terms may not be represented within the embeddings (e.g. representing "the answer is 4.3333" with the same embedding as, for example, "the answer is 2.987" if neither of the numbers are recognized). Particularly in mathematics contexts, such non-numeric terms are likely to greatly inform the correctness of the student response. As such, one possible solution is to expand the embedding space to include such terms; while plausible, this would likely require large datasets of mathematics responses, but even then would not be able to represent every possible number or expression (given that these are infinite).

Instead, we propose the "Math Term Frequency" (MTF) method which takes a much simpler approach, drawing inspiration from assessment methods applied for close-ended problems. The goal of this method is to supplement the previously-developed SBERT-Canberra model through ensembling, resulting in what we are calling the "SBERT-MTF" model, as illustrated in Figure 1.

The MTF method works by first parsing student answers to identify non-linguistic terms. The function[2] works through a sequence of steps, which includes splitting a student answer by spaces, removing alphabet-only terms (accounting for commas, uncommon punctuation, and contractions), combining equations separated by spaces, removing extraneous parentheses, and rounding off dec-

[2] All code used in this work is available at *url blinded for review*

imals, among other optimizations. Much of this pruning is done with regular expressions.

Once the non-linguistic terms have been identified, the MTF method involves identifying the most frequently-occurring terms for each possible integer score as a means of learning a kind of rubric. It is hypothesized that the correct student answers are likely to exhibit a smaller number of certain terms, with lower scores exhibiting a larger variety of terms. There will likely be some terms that are common throughout all scored answers (e.g. if the students reference a number from the problem text), but there are likely to be some terms that demonstrate high comprehension; similarly, students exhibiting common misconceptions may arrive at a similar set of incorrect answers. With this in mind, we select the five most-frequent terms, a_1, a_2, a_3, a_4, a_5, from the list of parsed non-linguistic terms. For each student response s in the training data for problem p, let S denote the set of mathematical terms in s. The input associated with s is the 5-vector $\langle \mathbb{1}_S(a_1), \mathbb{1}_S(a_2), \mathbb{1}_S(a_3), \mathbb{1}_S(a_4), \mathbb{1}_S(a_5) \rangle$, where $\mathbb{1}_S$ denotes the indicator function for a term in s. In other words, features used in this method indicate whether a newly-observed student response contains any of the most frequent terms most commonly associated with each given score. These features are used in a multinomial logistic regression (treating each score as an independent category, following previous works) that is trained separately for each problem.

The score predictions from the MTF model are then ensembled with the SBERT-Canberra predictions using another logistic regression model, referred to as the SBERT-MTF model; to clarify, this ensemble regression model observes ten features corresponding to the probability estimates produced for each of the five possible scores for each of the two observed models. The goal of this is to combine the semantic representation captured by the SBERT method, while taking advantage of the non-linguistic term matching from the MTF method.

4 SBERT-MTF Model Performance

As to directly compare the existing method to the prior works, we use similar evaluation method and dataset used in [2, 1]. This evaluation method utilizes a 2-parameter Rasch model to compare model estimates. The model predictions are used as covariates within the Rasch model [31] which additionally learns a parameter representing student ability and another for problem difficulty (commonly used in item response theory, or IRT, models in educational measurement); the number of words in the response is also added as a covariate in this evaluation model in an attempt to further compare models on their ability to interpret student answers rather than be based on other more superficial response features. This evaluation method allows for a fair comparison that accounts for factors that likely impact score that are external to the observed text of the student response. For comparison to previous works, we evaluate our method using three metrices: AUC score (calculated as an average AUC over each individual score category similar to [15]), Root Mean Squared Error (RMSE; calculated using

model estimates as a continuous-valued integer scale), and multi-class Cohen's Kappa.

The Rasch model performance of the Math terms frequency model as compared to the performance of the prior models for scoring open-ended responses is presented in Table 1. The results suggests that the proposed SBERT-MTF model outperforms the previous highest-performing SBERT-Canberra model across all three evaluation metrics.

Table 1. Rasch Model Performance compared to the models developed in prior works related to auto-scoring of student open responses in mathematics.

Model	AUC	RMSE	Kappa
Current Paper			
Rasch* + SBERT-MTF	**0.871**	**0.524**	**0.508**
Prior Works			
Baseline Rasch	0.827	0.709	0.370
Rasch* + Random Forest	0.850	0.615	0.430
Rasch* + SBERT-Canberra	0.856	0.577	0.476

*These rasch models also included the number of words.

5 Error Analysis of SBERT-MTF

The proposed MTF method was designed to address a very targeted problem exhibited by the previously-developed SBERT-Canberra model. We therefore conduct a similar error analysis to observe whether this method impacts the observed positive correlation between the presence of non-linguistic terms and model error. For this analysis, we use the second dataset as described in Section 3.1 for a direct comparison with the previous work. While the modeling task treats scoring as a categorization task, we convert the model predictions to a continuous-valued integer value (i.e. 0-4). We calculate model prediction error as the absolute value of the teacher-provided score (treated as ground truth) minus the predicted score. In this way, positive values correspond with higher error and values close to 0 represent low error (high performance). We calculate the answer-level features as introduced in Section 3.1 and conduct a linear regression observing absolute error as the dependent variable.

We compare three models within this analysis to identify how two modeling decisions presented in this work correspond with observed changes in feature coefficients. The first model observed is that of the SBERT-Canberra model reported in [2] as a baseline for comparison. The second model uses the same SBERT-Canberra method, but trains a logistic regression per problem with the model predictions as covariates (e.g. similar to the ensembled method described earlier, without MTF); the intuition here is that problem-specific adjustments

Table 2. The resulting model coefficients for the uni-level linear regression model of absolute error for SBERT Canberra, Logistic SBERT and MTF model.

	SBERT-Canberra		Logistic SBERT		SBERT-MTF	
	B	Std. Error	B	Std. Error	B	Std. Error
Intercept	0.581***	0.017	0.738***	0.017	0.776***	0.070
Answer Length	-0.008***	0.001	-0.008***	0.001	-0.009***	0.001
Avg. Word Length	-0.014***	0.003	-0.013***	0.003	-0.014***	0.003
Numbers Count	<0.001	<0.001	<0.001	<0.001	<0.001	<0.001
Operators Count	-0.006***	0.001	0.001	0.001	0.004**	0.001
Equation Percent	0.443***	0.018	-0.062***	0.019	-0.128***	0.019
Presence of Images	2.248***	0.021	2.058***	0.022	2.018***	0.022

*p <0.05 **p<0.01 ***p<0.001

may itself help to account for error in the model. Finally, we observe the ensembled SBERT-MTF model to observe any potential impacts beyond these other two methods.

The results of the error analysis is presented in Table 2. The results indicate that the linear model for both Logistic SBERT and SBERT-MTF explains 34.8% of the variance of the outcome as given by r-squared; this alone suggests that there is a large portion of variance in the error unexplained by the observed features. Among the observed features, similar to the results from [2], nearly all were statistically reliable in predicting the model error. However, it is arguable that from the relatively small scale of most coefficients, two of the features exhibit more meaningful impacts in comparison to the others: the presence of mathematical expression and presence of images in the student answers. However, with the introduction of a logistic regression model that follows the SBERT-Canberra method, the coefficient value of presence of mathematical terms has changed; it would appear that accounting for problem-level adjustments alone removes much of the impact of non-linguistic terms in the dataset. Most notably, however, is that the addition of our MTF method exhibits an even stronger negative correlation between the presence of non-linguistic terms and model error; what once was a weakness now appears to be a potential strength of the model.

6 Extrapolation of SBERT-MTF model To Student Responses in Algorithms

Following the improved performance observed from our SBERT-MTF method, we conduct a final analysis to explore how well this method extrapolates beyond the mathematics domain to a similar context where non-linguistic terms are common: computer science education. Similar to the mathematics domain, introductory computer science courses commonly observe open-ended problems where students utilize non-linguistic terms alongside natural language; if the

Table 3. Example of student responses given to open-ended questions in Algorithms class and their corresponding non-linguistic terms parsed using the developed parsing method.

Example	Non linguistic terms
O(n*logn)	[‘O(n*logn)’]
Because heapSort starts from the bottom up & doesn’t take a[0] and put it at a[0], it starts it at a[1] with sink() and swim() methods.	[‘&’, ‘a[0]’, ‘a[0]’, ‘a[1]’]
”if (left > mid) { a[k] = aux[right++]; }” means that if the left still has elements, an element from the auxiliary array will be added to the array a.	[‘>’, ‘{’, ‘a[k]=}’]

Table 4. Model Performance of Ensembled SBERT-MTF model applied on the dataset of student open responses in Algorithms class.

Model	AUC	RMSE	Kappa
SBERT-Canberra	0.691	0.424	0.304
SBERT-MTF	**0.711**	**0.408**	**0.364**

model extrapolates well to this new context, it suggests that the benefits of using our simple MTF method could help improve automated assessment models in a range of other contexts as well.

We took a dataset of student open-responses from an undergraduate-level Algorithms course. This dataset consists of 1,802 student responses to 13 different computer algorithm problems from an introductory Algorithms class taught to undergraduate students at the BLINDED UNIVERSITY. The average length of student responses in this dataset is about 61.21 words, with an average of 20% of these being non-linguistic terms. A few example of the student responses with the extracted non linguistic terms from this dataset is presented in Table 3.

We applied a 10-fold cross validation to get the predictions based on both the SBERT-Canberra method and the SBERT-MTF models and calculated the same three metrics as used in the prior analysis (note that this did not use the Rasch model evaluation). The results of these methods are presented in Table 4. Based on the results, Math terms frequency model outperforms the SBERT-Canberra method across all three metrices.

7 Discussion and Future Work

The results of all of the presented analyses illustrate MTF (specifically, SBERT-MTF) as a promising method to mitigate model error attributed to the presence of non-linguistic terms. The MTF method represents an intentionally-simple ap-

proach to address a targeted weakness observed in previously-developed models and seemingly led to positive impacts.

With that, there are still several areas in which these models could be improved, in addition to improving the accuracy of the parsing function. Most notably, is the remaining correlation between the presence of images and model error. While this is not surprising, as the models do nothing to account for images, this remains an unhandled case that cannot be ignored (i.e. student answers sometimes contain images, and current results suggest that this may lead to differential model performance across students more inclined to include images). As it is also the case that some students include mixtures of natural language, non-linguistic terms, and images all in the same answer, developing methods to handle such cases fairly is an important problem that is not addressed in the current work.

Similarly, the error analysis suggests that there is a large amount of variance in model error left unexplained. Previous work [2] identified problem- and teacher-level factors that seemingly account for much of this unexplained error, but this does not provide clear guidance as to how to account for these external factors fairly within an automatic assessment model.

The results of our method when applied to the Algorithms dataset provide promise for the generalization of these methods to domains beyond mathematics. As has been discussed, there are several fields where non-linguistic terms are common within student responses. This work highlights how state-of-the-art methods such as SBERT may be used in conjunction with simpler methods to draw from their respective strengths.

8 Conclusion

In this paper, we proposed an approach based on the occurrence of frequent math terminologies in student responses in predicting scores to student open-responses, and upon combining this approach with the prior works based on SBERT-Canberra model, we were able to see improvements in the models performance across all observed metrics. Through an observed error analysis, we were able to show that the proposed model improves specifically in the area of equation percentage. Further, we looked into the applicability of this method by extending it to a new domain similar to mathematics of college-level Algorithms class. We found that this method outperformed the existing state-of-the-art method in this new domain of Algorithms. As, such this method could be further expanded to be included in other domains that exhibit similar non-linguistic terms. Similar methods could be extended in auto-assessment of open-responses to provide formative feedback to students in an automated manner. We hope that this work acts as a step for future researchers towards how error analyses can be leveraged to address model limitations and further improve machine learning models.

9 Acknowledgements

Blinded for Review

References

1. Author. Blinded for review.
2. Author. Blinded for review.
3. Author. Blinded for review.
4. Sumit Basu, Chuck Jacobs, and Lucy Vanderwende. Powergrading: a clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics*, 1:391–402, 2013.
5. Michael Brooks, Sumit Basu, Charles Jacobs, and Lucy Vanderwende. Divide and correct: Using clusters to grade short answers at scale. In *Proceedings of the first ACM conference on Learning@ scale conference*, pages 89–98, 2014.
6. Jill Burstein, Claudia Leacock, and Richard Swartz. Automated evaluation of essays and short answers. 2001.
7. Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, et al. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*, 2018.
8. Eva Chen, Margaret Heritage, and John Lee. Identifying and monitoring students' learning needs with technology. *Journal of Education for Students Placed at Risk*, 10(3):309–332, 2005.
9. Hongbo Chen and Ben He. Automated essay scoring by maximizing human-machine agreement. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1741–1752, 2013.
10. Albert T Corbett and John R Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.
11. Laurie Cutrone, Maiga Chang, et al. Auto-assessor: computerized assessment system for marking student's short-answers automatically. In *2011 IEEE International Conference on Technology for Education*, pages 81–88. IEEE, 2011.
12. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
13. Semire Dikli. An overview of automated scoring of essays. *The Journal of Technology, Learning and Assessment*, 5(1), 2006.
14. G-H Gweon, Hee-Sun Lee, Chad Dorsey, Robert Tinker, William Finzer, and Daniel Damelin. Tracking student progress in a game-like learning environment with a monte carlo bayesian knowledge tracing model. In *Proceedings of the Fifth International Conference on Learning Analytics And Knowledge*, pages 166–170, 2015.
15. David J Hand and Robert J Till. A simple generalisation of the area under the roc curve for multiple class classification problems. *Machine learning*, 45(2):171–186, 2001.
16. Kenneth Holstein, Gena Hong, Mera Tegene, Bruce M McLaren, and Vincent Aleven. The classroom as a dashboard: Co-designing wearable cognitive augmentation for k-12 teachers. In *Proceedings of the 8th international conference on learning Analytics and knowledge*, pages 79–88, 2018.

17. Giuseppe Jurman, Samantha Riccadonna, Roberto Visintainer, and Cesare Furlanello. Canberra distance on ranked lists. In *Proceedings of advances in ranking NIPS 09 workshop*, pages 22–27. Citeseer, 2009.
18. Andrew S Lan, Divyanshu Vats, Andrew E Waters, and Richard G Baraniuk. Mathematical language processing: Automatic grading and feedback for open response mathematical questions. In *Proceedings of the Second (2015) ACM Conference on Learning@ Scale*, pages 167–176, 2015.
19. Claudia Leacock and Martin Chodorow. C-rater: Automated scoring of short-answer questions. *Computers and the Humanities*, 37(4):389–405, 2003.
20. Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Hui Xiong, Yu Su, and Guoping Hu. Ekt: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*, 33(1):100–115, 2019.
21. Tahira Mahboob, Sadaf Irfan, and Aysha Karamat. A machine learning approach for student assessment in e-learning using quinlan's c4. 5, naive bayes and random forest algorithms. In *2016 19th International Multi-Topic Conference (INMIC)*, pages 1–8. IEEE, 2016.
22. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
23. Michael Mohler and Rada Mihalcea. Text-to-text semantic similarity for automatic short answer grading. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 567–575, 2009.
24. Philip I Pavlik Jr, Hao Cen, and Kenneth R Koedinger. Performance factors analysis–a new alternative to knowledge tracing. *Online Submission*, 2009.
25. Shuai Peng, Ke Yuan, Liangcai Gao, and Zhi Tang. Mathbert: A pre-trained model for mathematical formula understanding. *arXiv preprint arXiv:2105.00377*, 2021.
26. Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.
27. Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.
28. Brian Riordan, Andrea Horbach, Aoife Cahill, Torsten Zesch, and Chungmin Lee. 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, 2017.
29. Jia Tracy Shen, Michiharu Yamashita, Ethan Prihar, Neil Heffernan, Xintao Wu, Ben Graff, and Dongwon Lee. Mathbert: A pre-trained language model for general nlp tasks in mathematics education. *arXiv preprint arXiv:2106.07340*, 2021.
30. Shashank Srikant and Varun Aggarwal. A system to grade computer programming skills using machine learning. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1887–1896, 2014.
31. Benjamin D Wright. Solving measurement problems with the rasch model. *Journal of educational measurement*, pages 97–116, 1977.
32. Lishan Zhang, Yuwei Huang, Xi Yang, Shengquan Yu, and Fuzhen Zhuang. An automatic short-answer grading model for semi-open-ended questions. *Interactive learning environments*, 30(1):177–190, 2022.
33. Siyuan Zhao, Yaqiong Zhang, Xiaolu Xiong, Anthony Botelho, and Neil Heffernan. A memory-augmented neural model for automated grading. In *Proceedings of the fourth (2017) ACM conference on learning@ scale*, pages 189–192, 2017.