Fooling Automatic Short Answer Grading Systems

Anna Filighera (✉)[1][0000−0001−5519−9959], Tim Steuer[1][0000−0002−3141−712X], and Christoph Rensing[1]

Technical University of Darmstadt, Multimedia Communications Lab, Germany
{anna.filighera,tim.steuer,christoph.rensing}@kom.tu-darmstadt.de

Abstract. With the rising success of adversarial attacks on many NLP tasks, systems which actually operate in an adversarial scenario need to be reevaluated. For this purpose, we pose the following research question: *How difficult is it to fool automatic short answer grading systems?* In particular, we investigate the robustness of the state of the art automatic short answer grading system proposed by Sung et al. towards cheating in the form of *universal adversarial trigger* employment. These are short token sequences that can be prepended to students' answers in an exam to artificially improve their automatically assigned grade. Such triggers are especially critical as they can easily be used by anyone once they are found. In our experiments, we discovered triggers which allow students to pass exams with passing thresholds of 50% without answering a single question correctly. Furthermore, we show that such triggers generalize across models and datasets in this scenario, nullifying the defense strategy of keeping grading models or data secret.

Keywords:

1   Introduction

Adversarial data sample perturbations, also called adversarial examples, intending to fool classification models have been a popular area of research in recent years. Many state of the art (SOTA) models have been shown to be vulnerable to adversarial attacks on various data sets [8, 44, 47].

On image data, the extent of modifications needed to change a sample's classified label are often so small they are imperceptible to humans [2]. On natural language data, perturbations can more easily be detected by humans. However, it is still possible to minimally modify samples so that the semantic meaning does not change but the class assigned by the model does [3, 6, 13, 17, 22, 29, 30].

While the existence of such adversarial examples unveils our models' shortcomings in many fields, they are especially worrying in settings where we actually expect to face adversaries. In this work, we focus on one such setting: automatic short answer grading (ASAG) systems employed in exams. ASAG systems take free-text answers and evaluate their quality with regards to their semantic content, completeness and relevance to the answered question. These

free-text answers are provided by students and are typically somewhere between a phrase and a paragraph long.

The willingness of college students to cheat has been well-studied [1, 9, 11, 18, 36, 37]. And while the exact percentage of cheating students varies greatly from study to study, Whitley [42] reports a mean of 43.1% of students cheating on examinations over 36 studies in his review. Klein et al.[19] report similar values for cheating on exams in their large scale comparison of cheating behaviors in different schools.

In these studies cheating behavior included copying from other students, getting the exam questions beforehand or bringing a cheat sheet to the exam. We argue that exploiting weaknesses in automatic grading schemes is just another, albeit less explored, form of cheating and expect the students' willingness to exhibit such behavior to be similar. Therefore, if we wish to employ automated grading systems in exams, we should ensure that the perceived cost of cheating them outweighs the benefits.

The perceived cost of cheating is made up of various factors, such as the punishment when caught, moral considerations or the difficulty of cheating in the first place [26]. In this work, we aim to investigate the last factor: *How difficult is it to fool automatic short answer grading systems?*

For this purpose, we first reproduce the SOTA approach to ASAG [39] which is based on the transformer model BERT [10]. Then we subject the reproduced models to adversarial attacks. In particular, we employ and modify the universal trigger attack proposed by Wallace et al. [41]. It generates short sequences of tokens, called *universal adversarial triggers*, which try to make a model predict a target class regardless of the actual sample.

In our context, students could prepend such a universal trigger targeted at the *correct* class to all of their answers in an exam to artificially improve their score. An example of such a trigger can be seen in Table 1. This kind of universal trigger attack is especially critical as such triggers can be easily employed by anyone once they are found.

In this work we make the following novel contributions:

 – Application of SOTA natural language processing insights to the educational scenario of exam grading
 – Modification of Wallace et al.'s universal trigger attack to make it more targeted at a specific class, namely the *correct* class
 – Investigation of trigger transferability across datasets and models

2 Related Work

Two research areas are relevant for our work: automatic short answer grading and adversarial attacks.

Table 1. An example showing how prepending the trigger sequence "none exits" to a student answer - taken from SCIENTSBANK's question EM-21b's unseen answers [12] - changes the prediction from *incorrect* to *correct*.

Question:	Georgia found one brown mineral and one black mineral. How will she know which one is harder?
Reference Answer:	The harder mineral will leave a scratch on the less hard mineral. If the black mineral is harder, the brown mineral will have a scratch.
Student Answer:	that it stay the scratch. → *incorrect*
Triggered Answer:	none exits that it stay the scratch. → *correct*

2.1 Adversarial Attacks in NLP

Adversarial attacks can be categorized into input dependent and input independent attacks. Input dependent attacks aim to modify **specific** inputs so that the model misclassifies them. Strategically inserting, deleting or replacing words with their synonyms [29], their nearest neighbors in the embedding space [3] or other words which have a high probability of appearing in the same context [47] are examples of such an attack. Samanta and Mehta [35] also consider typos which in turn result in valid words, e.g. goods and good, for their replacement candidate pool. Modifications can also be made on the character level by inserting noise, such as swapping adjacent characters or completely scrambling words [6]. Finally, the text can also be paraphrased to change the syntactic structure [17].

Input agnostic attacks, on the other hand, aim to find perturbations that lead to misclassifications on **all** samples. For instance, this can be done by selecting a single perturbation in the embedding space which is then applied to all tokens indiscriminately [15]. Alternatively, Ribeiro et al. [30] propose an approach that first paraphrases specific inputs to find *semantically equivalent adversaries* and then generalizes found examples to universal, *semantically equivalent adversarial rules*. Rules are selected to maximize semantic equivalence when applied to a sample, induce as many misclassifications as possible and are finally vetted by humans. An example of such a rule is "What is" → "What's".

Another input independent approach involves concatenating a series of adversarial words - triggers - to the beginning of every input sample [5]. The universal trigger attack [41] utilized in this work also belongs to this category. In Section 4 the attack is explained in more detail. Additional information on adversarial attacks can also be found in various surveys [44, 48].

2.2 Automatic Short Answer Grading

Systems that automatically score student answers have been explored for multiple decades. Proposed approaches include clustering student answers into groups of similar answers and assigning grades to whole clusters instead of individual

answers [4, 16, 45, 46], grading based on manually constructed rules or models of ideal answers [21, 43] and automatically assigning grades based on the answer's similarity to given reference answers. We will focus on similarity-based approaches here because most recent SOTA results were obtained using this kind of approach. However, more information on other approaches can be found in various surveys [7, 14, 32].

The earlier similarity-based approaches involve manually defining features that try to capture the similarity of answers on multiple levels [12, 24, 25, 33, 34, 38]. Surface features, such as lexical overlap or answer length ratios, are utilized by most feature engineered approaches. Semantic similarity measures are also common, be it in the form of sentence embedding distances or measures derived from knowledge bases like WordNet [28]. Some forms of syntactic features are also often employed. Dependency graph alignment or measures based on the part-of-speech tags' distribution in the answers would be examples of syntactic features. A further discussion of various features can be found in [27].

More recently, deep learning methods have also been adapted to the task of automatic short answer grading [20, 31, 40]. The key difference to the feature engineered approaches lies in the fact that the text's representation in the feature space is learned by the model itself. The best performing model (in terms of accuracy and F1 score) on the benchmark 3-way SemEval dataset [12] was proposed by Sung et al. [39]. They utilize the uncased BERT base model [10] which was pre-trained on the BooksCorpus [49] and the English Wikipedia. It was pre-trained on the tasks of predicting randomly masked input tokens and whether a sentence is another's successor or not. Sung et al. then fine-tune this deep bidirectional language representation model to predict whether an answer is *correct, incorrect* or *contradictory* compared to a reference answer. For this purpose, they append a feed-forward classification layer to the BERT model. The authors claim that their model outperforms even human graders.

3 Reproduction of SOTA ASAG Model

To reproduce the results reported by Sung et al. [39], we trained 10 models with the hyperparameters stated in the paper. Unreported hyperparameters were selected close to the original BERT model's values with minimal tuning. The models were trained on the shuffled training split contained in the SCIENTSBANK dataset of the SemEval-2013 challenge. As in the reference paper, we use the 3-way task of predicting answers to be *correct, incorrect* or *contradictory.* Then the models were evaluated on the test split. The test set contains three distinct categories: unseen answers, unseen questions and unseen domains. Unseen answers are answers to questions for which some answers have already been seen during training. Unseen questions are completely new questions and the unseen domains category contains questions belonging to domains the model has not seen during training.

We were not able to reproduce the reported results exactly with this setup. Out of the 10 models, Model 4 and 8 performed best. A comparison of their

and the reported model's results can be seen in Table 2. The 10 models' average performance can be seen in Figure 1. Since the reported results are mostly within one or two standard deviations of the results achieved in our experiments, more in-depth hyperparameter tuning and reruns with different random seeds may yield the reported results. Alternatively, the authors may have taken steps that they did not discuss in the paper. However, as this is not the focus of this work, we deemed the reproduced models sufficient for our experiments.

Table 2. Performance of best reproduced models, Model 4 and 8, compared to the results reported by Sung et al. [39] in terms of accuracy (Acc), macro-averaged F1 score (M-F1) and weighted-averaged F1 score (W-F1). Each category's best result is marked in bold.

	Unseen Answers			Unseen Questions			Unseen Domains		
	Acc	M-F1	W-F1	Acc	M-F1	W-F1	Acc	M-F1	W-F1
#4	0.744	0.703	0.741	*0.675*	0.555	*0.665*	0.624	0.490	0.609
#8	0.737	0.690	0.732	0.674	0.561	0.662	*0.670*	*0.599*	*0.661*
Ref.	*0.759*	*0.720*	*0.758*	0.653	*0.575*	0.648	0.638	0.579	0.634

4   Universal Trigger Attack

In this work, we employ the universal trigger attack proposed by Wallace et al. [41]. It is targeted, meaning that a target class is specified and the search will try to find triggers that lead the model to predict the specified class, regardless of the sample's actual class. The attack begins with an initial trigger, such as "the the the", and iteratively searches for good replacements for the words contained in the trigger. The replacement strategy is based on the HotFlip attack proposed by Ebrahimi et al. [13]. For each batch of samples, candidates are chosen out of all tokens in the vocabulary so that the loss for the target class is minimized. Then, a beam search over candidates is performed to find the ordered combination of triggers which maximizes the batch's loss.

We augment this attack by also considering more target label focused objective functions for the beam search than the batch's loss. Namely, we experiment with naively maximizing the number of target label predictions and the targeted LogSoftmax function depicted in Equation 1. Here, $L = \{$*correct, incorrect, contradictory*$\}$, t is the target label, n denotes the number of samples in the batch $\mathbf{x}$ and $f_l(x)$ represents the model's output for label l's node before the softmax activation function given a sample x.

$$TargetedLogSoftmax(t, \mathbf{x}) = \sum_{i=0}^{n} log \left(\frac{exp(f_t(x_i))}{\sum_{j \epsilon L} exp(f_j(x_i))} \right) \tag{1}$$
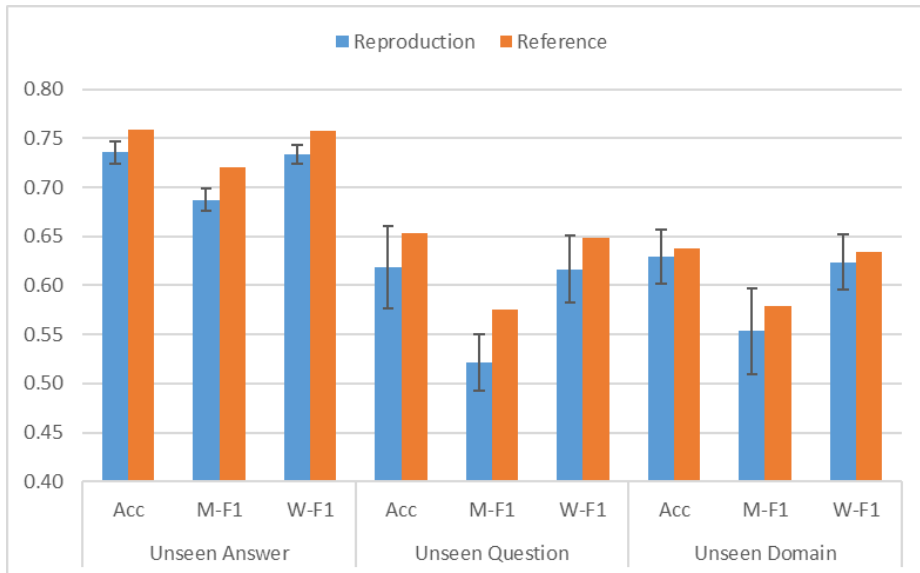
Fig. 1. Average performance of the 10 reproduction models compared to the results reported by Sung et al. [39].The black bars represent one standard deviation in each direction. Please note that the y axis begins at 0.4 instead of 0.

5 Experiments

In this section, we first give a short overview of the datasets used in our experiments. Then, we present the best triggers found, followed by a short investigation of the effect of trigger length on the number of successful flips. Next, the effect of our modified objective function is investigated. Finally, we report on the transferability of triggers across models.

5.1 Data

The SemEval ASAG challenge consists of two distinct datasets: SCIENTSBANK and BEETLE. While the BEETLE set only contains questions concerning electricity, the SCIENTSBANK corpus includes questions of various scientific domains, such as biology, physics and geography. We do not include the class distribution of the 3-way task here, as it can be found in the original [12] and the ASAG reference paper [39].

5.2 Experiment Setup

Unless explicitly stated, all experiments were conducted in the following way. Model 8 was chosen as the victim model because it has the overall best performance of all reproduction models. See Table 2 for reference. Since the model was

trained on the complete SCIENTSBANK training split as stated in the reference paper, we selected the BEETLE training split as basis for our attacks. While the class labels were homogenized for both datasets in the SemEval challenge, the datasets are still vastly different. They were collected in dissimilar settings, by different authors and deal with disparate domains [12]. **This is important, as successful attacks with this setup imply transferability of triggers across datasets**. In practice, this would allow attackers to substitute secret datasets with their own corpora and still find successful attacks on the original data. To the best of our knowledge, experiments investigating the transferability of natural language triggers across datasets are a novel contribution of our work.

From the BEETLE training set all 1227 *incorrect* samples were selected. The goal of the attack was to flip their classification label to *correct*. We would also have liked to try and flip *contradictory* examples. However, the model was only able to correctly predict 18 of the 1049 *contradictory* samples without any malicious intervention necessary. Finally, the triggers found are evaluated on the SCIENTSBANK test split.

5.3   Results

In the related work, the success of an attack is most often measured in the drop in accuracy it is able to achieve. However, this would overestimate the performance in our scenario as we are only interested in *incorrect* answers which are falsely graded as *correct* in contrast to answers which are labeled as *contradictory*. Therefore, we also report the absolute number of successful flips from *incorrect* to *correct*.

During the iterative trigger search process described in Section 4 a few thousand triggers were evaluated on the BEETLE set. Of these, the 20 triggers with the most achieved flips were evaluated on the test set and of these, the best triggers can be seen in Table 3. On the unseen answers test split, the model without any triggers misclassified 12.4% (31) of all *incorrect* samples as *correct*. The triggers "none varies" and "none would" managed to flip an additional 8.8% of samples so that 21.3% (53) are misclassifed in total. On the unseen questions split, the base misclassification rate was 27.4% (101) and "none would" added another 10.1% for a total of 37.5% (138). On the unseen domains split, "none elsewhere" increased the misclassification rate from 22.0% (491) to 37.1% (826).

Effect of Trigger Length Wallace et al. [41] state that the trigger length is a trade-off between effectiveness and stealth. They experimented with prepending triggers of lengths between 1 and 10 tokens and found longer triggers to have higher success rates. This differs from observations made in our experiments. When the *correct* class is targeted, a trigger length of two achieves the best results, as can be seen in Table 3. On the unseen answers split, the best trigger of length 3 is "heats affected penetrated" and it manages to flip only 42 samples.

Table 3. The triggers with the most flips from *incorrect* to *correct* for each test split. The number of model 8's misclassifications without any triggers can be found in the last row. For the sake of comparability with related work, the accuracy for *incorrect* samples is also given. UA stands for "unseen answers", UQ denotes "unseen questions" and UD represents "unseen domains".

	Number of Flips			Accuracy		
Triggers	UA	UQ	UD	UA	UQ	UD
none varies	53	134	687	71.08	54.62	63.69
none would	53	138	810	41.77	31.25	31.15
none elsewhere	50	121	826	47.79	36.14	37.93
Base misclassification	31	101	491	84.74	70.65	76.93

The number of successful flips further decreases to 9 for the best trigger of length 4, "##ired unaffected least being". The same trend also holds for the other test splits but is omitted here for brevity. This difference in observation may be due to the varying definitions of attack success. Wallace et al. [41] view a trigger as successful as soon as the model assigns **any** class other than the true label, while we only accept triggers which cause a prediction of the class *correct*. The educational setting of this work may also be a factor.

Effect of Objective Function We compared the performance of the three different objective functions described in Section 4, namely the original function proposed by Wallace et al. [41], the targeted LogSoftmax depicted in Equation 1 and the naive maximization of the number of target label predictions. To make the comparison as fair as possible while keeping the computation time reasonable, we fixed the hyperparameters of the attack to a beam size of 4 and a candidate set size of 100. The attack was run for the same number of iterations exactly once for each function. The best triggers found by each function can be seen in Table 4. The performance is relatively similar, with the targeted function achieving the most flips on all test splits, closely followed by the original function and, lastly, the naive prediction maximization. Qualitative observation of all produced triggers showed that the original function's triggers resulted in more flips from *incorrect* to *contradictory* than the proposed targeted function's.

Table 4. A comparison of the objective functions.

		Number of Flips		
Objective Function	Best Trigger	UA	UQ	UD
Naive	none cause	42	107	647
Original	nobody penetrated	43	121	673
Targeted	none elsewhere	50	121	826

Transferability One of the most interesting aspects of triggers relates to the ability to find them on one model and use them to fool another model. In this setting, attackers do not require access to the original model, which may be kept secret in a grading scenario. Trigger transferability across models allows them to train a substitute model for the trigger search and then attack the actual grading model with found triggers. We investigate this aspect by applying all good triggers found on Model 8 to Model 4. Note that this also included triggers from a search on the SCIENTSBANK training split and not just the BEETLE training set. The best performing triggers in terms of flips induced in Model 4 can be seen in Table 5. We also included the triggers which performed best on Model 8 here.

Table 5. Performance of the triggers found on Model 8 evaluated on Model 4. For reference, the number of flips originally achieved on Model 8 are also given. The first rows are the best performing triggers on Model 4. The middle block contains the best triggers on Model 8. Finally, the last row gives the number of samples missclassified by Model 4 without any triggers.

| | Number of Flips on Model 4 and 8 | | | | | |
| | UA | | UQ | | UD | |
Trigger	4	8	4	8	4	8
nowhere changes	**81**	51	**184**	135	957	640
anywhere.	58	45	108	105	**1027**	682
none else	73	53	158	136	941	818
none varies	49	53	79	134	576	687
none would	38	53	97	**138**	495	810
none elsewhere	60	50	115	121	701	**826**
Base Missclasification	44	31	100	101	646	491

While there are triggers that perform well on both models, e.g. "none else", the best triggers for each model differ. Interestingly, triggers like "nowhere changes" or "anywhere." perform even better on Model 4 than the best triggers found for the original victim model. On UA, "nowhere changes" flips 14.9% of all *incorrect* samples to *correct*. In addition to the base misclassification rate, this leads to a misclassification rate of 32.5%. On UQ, the same trigger increases the misclassification rate by 22.8% to a total of 50%. On the UD split, prepending "anywhere." to all *incorrect* samples raises the rate by 17.1% to 46.1%.

As a curious side note, the trigger "heats affected penetrated" discussed in the section regarding trigger length performed substantially better on Model 4, so that it was a close contender for the best trigger list.

6   Discussion and Conclusion

In our scenario, a misclassification rate of 37.5% means that students using triggers only need to answer 20% of the questions correctly to pass a test that was designed to have a passing threshold of 50%. If an exam would be graded by Model 4, students could pass the test by simply prepending "nowhere changes" to their answers without answering a single question correctly! However, this does not mean that these triggers flip any arbitrary answer, as a large portion of the flipped *incorrect* answers showed at least a vague familiarity with the question's topic similar to the example displayed in Table 1. Additionally, these rates were achieved on the unseen questions split. Translated to our scenario this implies that we would expect our model to grade questions similar to questions it has seen during training but for which it has not seen a single example answer, besides the reference answer. To take an example out of the actual dataset, a model trained to grade the question *What happens to earth materials during deposition?* would also be expected to grade *What happens to earth materials during erosion?* with only the help of the reference answer "Earth materials are worn away and moved during erosion.". The results suggest that the current SOTA approach is ill-equipped to generalize its grading behavior in such a way.

Nevertheless, even if we supply training answers to every question the misclassification rates are quite high with 21.3% and 32.5% for Model 8 and 4, respectively. Considering how easy these triggers are employed by everyone once someone has found them, this is concerning. Thus, defensive measures should be investigated and put into place before using automatic short answer grading systems in practice.

In conclusion, we have shown the SOTA automatic short answer grading system to be vulnerable to cheating in the form of universal trigger employment. We also showed that triggers can be successful even if they were found on a disparate dataset or model. This makes the attack easier to execute, as attackers can simply substitute secret grading components in their search for triggers. Lastly, we also proposed a way to make the attack more focused on flipping samples from a specific source class to a target class.

7   Future Work

There are several points of interest which we plan to study further in the future. For one, finding adversarial attacks on natural language tasks is a very active field at the moment. Exposing ASAG systems to other forms of attacks, such as attacks based on paraphrasing, would be very interesting. Additionally, one could also explore defensive measures to make grading models more robust. An in-depth analysis of why these attacks work would be beneficial here. Finally, expanding the transferability study conducted in this work to other kinds of models, such as RoBERTa [23] or feature engineering-based approaches, and additional datasets may lead to interesting findings as well.

References

1. Ahmadi, A.: Cheating on exams in the iranian efl context. Journal of Academic Ethics **10**(2), 151–170 (2012)
2. Akhtar, N., Mian, A.: Threat of adversarial attacks on deep learning in computer vision: A survey. IEEE Access **6**, 14410–14430 (2018)
3. Alzantot, M., Sharma, Y., Elgohary, A., Ho, B.J., Srivastava, M., Chang, K.W.: Generating natural language adversarial examples. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. pp. 2890–2896 (2018)
4. Basu, S., Jacobs, C., Vanderwende, L.: Powergrading: a clustering approach to amplify human effort for short answer grading. Transactions of the Association for Computational Linguistics **1**, 391–402 (2013)
5. Behjati, M., Moosavi-Dezfooli, S.M., Baghshah, M.S., Frossard, P.: Universal adversarial attacks on text classifiers. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 7345–7349. IEEE (2019)
6. Belinkov, Y., Bisk, Y.: Synthetic and natural noise both break neural machine translation. arXiv preprint arXiv:1711.02173 (2017)
7. Burrows, S., Gurevych, I., Stein, B.: The eras and trends of automatic short answer grading. International Journal of Artificial Intelligence in Education **25**(1), 60–117 (2015)
8. Carlini, N., Wagner, D.: Adversarial examples are not easily detected: Bypassing ten detection methods. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. pp. 3–14. ACM (2017)
9. Danielsen, R.D., Simon, A.F., Pavlick, R.: The culture of cheating: From the classroom to the exam room. Journal of Physician Assistant Education (Physician Assistant Education Association) **17**(1) (2006)
10. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 4171–4186 (2019)
11. Diekhoff, G.M., LaBeff, E.E., Shinohara, K., Yasukawa, H.: College cheating in japan and the united states. Research in Higher education **40**(3), 343–353 (1999)
12. Dzikovska, M.O., Nielsen, R.D., Brew, C., Leacock, C., Giampiccolo, D., Bentivogli, L., Clark, P., Dagan, I., Dang, H.T.: Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. Tech. rep., NORTH TEXAS STATE UNIV DENTON (2013)
13. Ebrahimi, J., Rao, A., Lowd, D., Dou, D.: Hotflip: White-box adversarial examples for text classification. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 31–36 (2018)
14. Galhardi, L.B., Brancher, J.D.: Machine learning approach for automatic short answer grading: A systematic review. In: Ibero-American Conference on Artificial Intelligence. pp. 380–391. Springer (2018)
15. Gao, H., Oates, T.: Universal adversarial perturbation for text classification. arXiv preprint arXiv:1910.04618 (2019)
16. Horbach, A., Pinkal, M.: Semi-supervised clustering for short answer scoring. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018) (2018)

17. Iyyer, M., Wieting, J., Gimpel, K., Zettlemoyer, L.: Adversarial example generation with syntactically controlled paraphrase networks. In: Proceedings of NAACL-HLT. pp. 1875–1885 (2018)
18. King, C.G., Guyette Jr, R.W., Piotrowski, C.: Online exams and cheating: An empirical analysis of business students' views. Journal of Educators Online **6**(1), n1 (2009)
19. Klein, H.A., Levenburg, N.M., McKendall, M., Mothersell, W.: Cheating during the college years: How do business school students compare? Journal of Business Ethics **72**(2), 197–206 (2007)
20. Kumar, S., Chakrabarti, S., Roy, S.: Earth mover's distance pooling over siamese lstms for automatic short answer grading. In: IJCAI. pp. 2046–2052 (2017)
21. Leacock, C., Chodorow, M.: C-rater: Automated scoring of short-answer questions. Computers and the Humanities **37**(4), 389–405 (2003)
22. Liang, B., Li, H., Su, M., Bian, P., Li, X., Shi, W.: Deep text classification can be fooled. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence. pp. 4208–4215. AAAI Press (2018)
23. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
24. Marvaniya, S., Saha, S., Dhamecha, T.I., Foltz, P., Sindhgatta, R., Sengupta, B.: Creating scoring rubric from representative student answers for improved short answer grading. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. p. 993–1002. CIKM '18, Association for Computing Machinery, New York, NY, USA (2018). https://doi.org/10.1145/3269206.3271755, https://doi.org/10.1145/3269206.3271755
25. Mohler, M., Bunescu, R., Mihalcea, R.: Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. pp. 752–762. Association for Computational Linguistics (2011)
26. Murdock, T.B., Anderman, E.M.: Motivational perspectives on student cheating: Toward an integrated model of academic dishonesty. Educational psychologist **41**(3), 129–145 (2006)
27. Padó, U.: Get semantic with me! the usefulness of different feature types for short-answer grading. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. pp. 2186–2195 (2016)
28. Pedersen, T., Patwardhan, S., Michelizzi, J.: Wordnet:: Similarity: measuring the relatedness of concepts. In: Demonstration papers at HLT-NAACL 2004. pp. 38–41. Association for Computational Linguistics (2004)
29. Ren, S., Deng, Y., He, K., Che, W.: Generating natural language adversarial examples through probability weighted word saliency. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 1085–1097 (2019)
30. Ribeiro, M.T., Singh, S., Guestrin, C.: Semantically equivalent adversarial rules for debugging nlp models. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 856–865 (2018)
31. Riordan, B., Horbach, A., Cahill, A., Zesch, T., Lee, C.M.: 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 (2017)

32. Roy, S., Narahari, Y., Deshmukh, O.D.: A perspective on computer assisted assessment techniques for short free-text answers. In: International Computer Assisted Assessment Conference. pp. 96–109. Springer (2015)

33. Saha, S., Dhamecha, T.I., Marvaniya, S., Sindhgatta, R., Sengupta, B.: Sentence level or token level features for automatic short answer grading?: Use both. In: International Conference on Artificial Intelligence in Education. pp. 503–517. Springer (2018)

34. Sahu, A., Bhowmick, P.K.: Feature engineering and ensemble-based approach for improving automatic short-answer grading performance. IEEE Transactions on Learning Technologies (2019)

35. Samanta, S., Mehta, S.: Towards crafting text adversarial samples. arXiv preprint arXiv:1707.02812 (2017)

36. Sheard, J., Dick, M., Markham, S., Macdonald, I., Walsh, M.: Cheating and plagiarism: perceptions and practices of first year it students. In: ACM SIGCSE Bulletin. vol. 34, pp. 183–187. ACM (2002)

37. Smyth, M.L., Davis, J.R.: An examination of student cheating in the two-year college. Community College Review **31**(1), 17–32 (2003)

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

39. Sung, C., Dhamecha, T.I., Mukhi, N.: Improving short answer grading using transformer-based pre-training. In: International Conference on Artificial Intelligence in Education. pp. 469–481. Springer (2019)

40. Tan, C., Wei, F., Wang, W., Lv, W., Zhou, M.: Multiway attention networks for modeling sentence pairs. In: IJCAI. pp. 4411–4417 (2018)

41. Wallace, E., Feng, S., Kandpal, N., Gardner, M., Singh, S.: Universal adversarial triggers for attacking and analyzing nlp. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 2153–2162 (2019)

42. Whitley, B.E.: Factors associated with cheating among college students: A review. Research in higher education **39**(3), 235–274 (1998)

43. Willis, A.: Using nlp to support scalable assessment of short free text responses. In: Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications. pp. 243–253 (2015)

44. Yuan, X., He, P., Zhu, Q., Li, X.: Adversarial examples: Attacks and defenses for deep learning. IEEE transactions on neural networks and learning systems (2019)

45. Zehner, F., Sälzer, C., Goldhammer, F.: Automatic coding of short text responses via clustering in educational assessment. Educational and psychological measurement **76**(2), 280–303 (2016)

46. Zesch, T., Heilman, M., Cahill, A.: Reducing annotation efforts in supervised short answer scoring. In: Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications. pp. 124–132 (2015)

47. Zhang, H., Zhou, H., Miao, N., Li, L.: Generating fluent adversarial examples for natural languages. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 5564–5569 (2019)

48. Zhang, W.E., Sheng, Q.Z., Alhazmi, A., Li, C.: Adversarial attacks on deep learning models in natural language processing: A survey (2019)

49. Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., Fidler, S.: Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In: Proceedings of the IEEE international conference on computer vision. pp. 19–27 (2015)