Comparative Evaluation of Pretrained Transfer Learning Models on Automatic Short Answer Grading

Sasi Kiran Gaddipati, Deebul Nair and Paul G. Plöger
Hochschule Bonn-Rhein-Sieg, Germany
sasi-kiran.gaddipati@smail.inf.h-brs.de, {deebul.nair, paul.ploeger}@h-brs.de

Keywords: Automatic Short Answer Grading, Transfer Learning, ELMo, BERT, GPT, GPT-2

Abstract: The increasing number of students on digital platforms makes it difficult for the examiners and evaluators to test and grade the descriptive answers manually. Automatic Short Answer Grading (ASAG) is the process of grading the student answers automatically by computational approaches given a question and the desired answer. Previous works implemented the methods of concept mapping, facet mapping, and some used the conventional word embeddings for extracting semantic features. They extracted the multiple features manually to train on the corresponding datasets. We use the pretrained embeddings of the transfer learning models, ELMo, BERT, GPT, and GPT-2 to assess their efficiency on the task. We train with the single feature, cosine similarity, extracted from the embeddings of these models to evaluate on the Mohler dataset. We compare the RMSE scores and correlation measurements of the four models with former methods on Mohler dataset. Our work demonstrates that ELMo outperformed the other three models. We also, briefly describe the four transfer learning models and conclude with the possible causes of poor outcomes of transfer learning models.

1 INTRODUCTION

Descriptive answers test students' comprehensive understanding of topics. With the growing number of students on online platforms and universities, it is very strenuous to evaluate all the answers manually. The process of grading these detailed answers automatically without a human explanation is achieved by ASAG. ASAG, given a question and the desired answer, assesses, and grade the students' answer as shown in Fig. 1.

(Pérez et al., 2005), (Bukai et al., 2006), considered the corpus-based methods to evaluate the student answer. (Gütl, 2007), (Bailey and Meurers, 2008), (Hou and Tsao, 2011) introduced machine learning techniques into ASAG. (Mohler et al., 2011) have provided several syntactic, lexical, morphological, and semantic methods for ASAG. Previous works have manually combined some of the aforementioned language features to train on the regression model. One of such features is the semantic similarities between the desired answer and the student answer. This can be considered as the Semantic Text Similarity (STS) task, which assigns the similarity score between two textual corpora. The latest advancements in natural language processing and deep learning have provided propitious methods in the STS task. Some of these methods produced robust transfer learning models

The transfer learning models have been pretrained on huge corpora and are able to extract the semantic context of the words with robust architectures as we explain in Sec 3. These models include Embeddings from Language Models (ELMo) (Peters et al., 2018), Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018), Generative Pretraining (GPT) (Radford et al., 2018) and GPT-2 (Radford et al., 2019), which have shown the state-of-the-art results on various tasks. We use these models to extract the semantic knowledge from the answers by encoding with contextual vectors. We also use fewer preprocessing steps to train a regression model compared to previous works, to assess the potentiality of transfer learning models. We evaluate these models on the Mohler dataset.

Further, we discuss the related work in Section 2. Section 3 briefly explains the transfer learning models and Section 4 elaborates the dataset. Section 5 details the experimentation followed by the results in Section 6. We discuss our observations from the results on ASAG in Section 7 and conclude in the Section 8. Finally, Section 9 explains the future work.

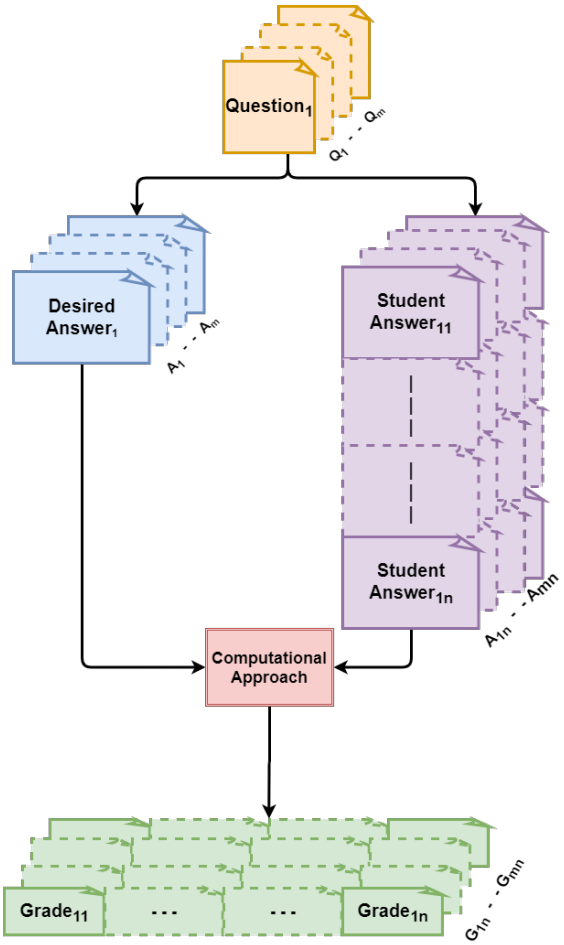

Figure 1: Automatic short answer grading pipeline.

2 RELATED WORK

(Burstein et al., 1999), (Callear et al., 2001), (Wang et al., 2008) worked with concept mapping techniques, by mapping the concepts of student answer with desired answer. Later the works of (Mitchell et al., 2002), (Bachman et al., 2002) (Thomas, 2003), extracted the information from the answers through pattern matching. They have used regular expressions and parse trees for extracting the patterns. (Pérez et al., 2005) introduced the corpus-based methods to ASAG with the combination of Latent Semantic Analysis (LSA) and Bi-lingual Evaluation Understudy (BLEU) scores. (Mohler and Mihalcea, 2009) used the combination of multiple knowledge based and corpus-based features for extracting the similarity between the students' and teacher's answer. This work is followed by the combining corpus-based methods into machine learning systems (Mohler et al., 2011).

(Sultan et al., 2016) implemented the term frequency–inverse document frequency (tf-idf) on Mohler dataset. (Metzler, 2019) used the conventional embeddings in Natural Language Processing (NLP) such as Word2Vec (Mikolov et al., 2013b) (Mikolov et al., 2013a), GloVe (Pennington et al., 2014) and FastText (Bojanowski et al., 2017) to extract the distributional and semantic features through embeddings. (Metzler, 2019) compared the results with pretrained word embeddings and domain-specific trained word embeddings of Word2Vec, GloVe and FastText on Mohler dataset.

However, the conventional embeddings did not consider the context of the words and long-term dependencies. The later advancements in NLP contemplated the contexts of the words in the sentences, pretrained on huge corpora. This resulted in the models like ELMo (Peters et al., 2018), GPT (Radford et al., 2018), BERT (Devlin et al., 2018) and GPT-2 (Radford et al., 2019). In addition, the former works tend to use multiple features to evaluate the answers. We use only a single semantic feature, extracted from the transfer learning models to train.

3 TRANSFER LEARNING MODELS

We use the embeddings of the pre-trained transfer learning models to extract the semantics of the words based on their context. These transfer learning models are initially trained on a source task and can be used to various target tasks by finetuning the ultimate layers. However, we do not consider the idea of transfer learning, instead we only use the pretrained embeddings of these models trained on the source task, assuming that they have extracted the significant features of each word from the pretrained huge corpora. The brief explanation of the transfer learning models in creating their pretrained embeddings are explained in the subsequent discussions.

ELMo is built with three layers of Long Short-Term Memory (LSTM), similar to that used by (Jozefowicz et al., 2016) (Kim et al., 2016) for complex language modeling. ELMo calculates the bi-directional context of the words by concatenating the joint probabilities of forward language model and backward language model. This assists highly for homonyms[1] such as *play*, *train* and *spring*. ELMo assigns different words embeddings for the same word in different contexts. The final layers of stacked LSTM are good at retriev-

[1]Homonyms are the words having same spelling and pronunciation but different meanings

Table 1: An overview of pretrained transfer learning models

Model	Architecture	Pretrained dataset	Dataset size
ELMo	Stacked bi-LSTM	One billion word benchmark	1B words
GPT	Stacked transformer	BookCorpus	800M words
BERT	Stacked transformer	BookCorpus English Wikipedia	800M words 2500M words
GPT2	Stacked transformer	WebText	*Not known*

ing semantics, while the initial layers are better at extracting syntactic information.

GPT uses a stacked transformer architecture (Vaswani et al., 2017) to train the weights. Stacked transformer architecture provided more structured memory for the long-term dependencies compared to recurrent neural networks (Radford et al., 2018). GPT aims to create an effective procedure for transfer learning through semi-supervised approach, with a combination of unsupervised pretraining and supervised finetuning. Principally, a language model objective is applied as a source task on unlabeled data, to learn initial parameters of the neural network. Later, the architecture is finetuned for a required task using corresponding supervised traversal-style approaches (Rocktäschel et al., 2015). The traversal-style approach creates a single contiguous sequence of tokens for a structured text.

BERT extracts the benefits of both ELMo and GPT, resulting in using the transformer mechanism (Vaswani et al., 2017) and capturing bi-directional context. BERT is trained on a multi-layer bidirectional transformer encoder with a huge corpus of BookCorpus (Zhu et al., 2015) and Wikipedia datasets. The authors have not adopted the traditional language modeling as source task, because it is only possible to train language models either left-to-right or right-to-left, which seemed to lose the essence of capturing the bi-directional context effectively (Devlin et al., 2018). Instead, they introduced an approach called Masked Language Model (MLM) as a source task. MLM masks some tokens in the sentences and predicts the tokens during training.

GPT-2 aims to show that language models can learn multi-tasks through unsupervised learning. The former GPT model used a combination of unsupervised pretraining and supervised finetuning. Language modeling is the core part of GPT-2. The authors used the transformer architecture with the similar architecture of GPT with minimal changes. They extracted the WebText dataset from web scraping, considering it to be more generalized.

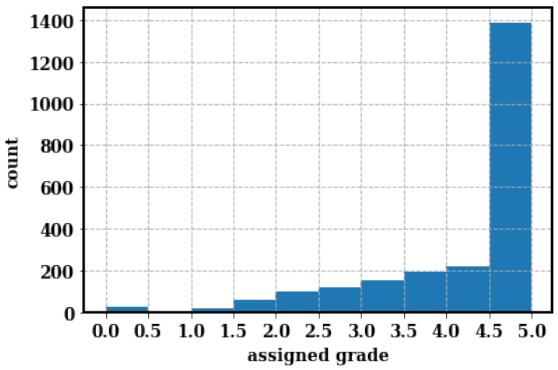

Figure 2: Histogram of assigned average scores in Mohler dataset representing the bias towards correct answers.

An overview of the four transfer learning models' architectures and datasets are represented in the Table 1.

4 DATASET

Mohler dataset comprises of questions and answers in computer science domain (Mohler et al., 2011). The goal of the dataset is to evaluate the model in grading the students' answers by comparing them with the evaluator's desired answer. It constitutes 2273 answers from 10 assignments and 2 examinations, collected from 31 students for 80 different questions.

Each answer in the assignment is graded from 0 (not correct) to 5 (totally correct) by two evaluators, who are specialized in the field of computer science. The average of the two evaluators' scores is considered as the standard score of each answer. The answers in examinations are graded from 0 (not correct) to 10 (totally correct). To eliminate the clutter of assigned scores in the dataset, (Metzler, 2019) had normalized all the grades of examinations to 0-5. Hence, we intend to use this cleaned data in our experimentation and evaluation.

The dataset is biased towards correct answers (Mohler et al., 2011) with the mean of average grades being 4.17 and median being 4.50 of the average scores. This bias can be inferred from the Fig. 2.

Table 2: Pearson correlation of pretrained transfer learning models on Mohler dataset

Model	Isotonic regression	Linear regression	Ridge regression
ELMo	**0.485**	**0.451**	**0.449**
GPT	0.248	0.222	0.217
BERT	0.318	0.266	0.269
GPT-2	0.311	0.274	0.269

Table 3: Root Mean Square Error (RMSE) score of pretrained transfer learning models on Mohler dataset

Model	Isotonic regression	Linear regression	Ridge regression
ELMo	**0.978**	**0.995**	**0.996**
GPT	1.082	1.088	1.089
BERT	1.057	1.077	1.075
GPT-2	1.065	1.078	1.079

5 EXPERIMENTATION

Preprocessing We have conducted only a tokenization in the preprocessing step. Lemmatization and stop word removal are neglected consciously, to analyze the performance of transfer learning models in raw setup. (Metzler, 2019) also used the spell checker for correcting the misspelled words, which we have avoided. We have assumed that the graders have deducted grades for the misspelled words in the students' answers. Henceforth, the existent spelling mistakes may be trained as a negative feature[2], internally. Since these transfer learning models are trained on huge vocabulary, it is plausible to assume that they can understand the misspelled words to an extent. The versatility of transfer learning models to assign an embedding to the new words also assisted in disregarding the spell mistakes.

Feature extraction The pretrained embeddings of each transfer learning model are assigned to the tokens of each word in all the answers. Considering there exists n_j words for an answer j of question i, we create answer embeddings by Sum of Word Embeddings (SOWE) as depicted in Eq 1, where a_{ij} represents the j^{th} answer vector of question q_i, w_k represents the vector of the k^{th} word in the answer a_{ij}. This creates a single vector representing each answer in high dimensional hypothesis space. The size of the sentence embeddings is equal to the size of the word embeddings.

$$a_{ij} = \sum_{j=1}^{n_j} w_k \tag{1}$$

We calculate the similarity between each student answer a_{ij} and desired answer a_i with cosine similarity given in Eq 2. We normalize these scores from 0 to 1 to scale the similarities and attain the relative measure of similarities. We consider these scores as the features of the answers and train them with different regression methods.

$$cos(a_{ij}, a_i) = \frac{a_{ij}.a_i}{|a_{ij}||a_i|} \tag{2}$$

Training and Testing We randomly split the Mohler data to 70% of training and 30% of testing data. We train each model for 1000 iterations selecting different training and testing data randomly for every iteration to generalize the results. We train the cosine similarity feature with the correspondingly assigned grades with isotonic, linear and non-linear (ridge) regressions. We implement the selected regression models to compare our results with (Mohler et al., 2011) and (Metzler, 2019). Followed by training, we test the trained regression model on test data. This test data is not seen by the regression model until it's testing phase. The test data's similarity scores are input through the trained regression model. This results in the predicted grades, which will be further used for evaluation. During evaluation, we calculate the RMSE and Pearson correlation between the predicted scores and desired scores.

6 RESULTS

Table 2 depicts the Pearson correlation (ρ) results of pretrained embeddings of transfer learning models on Mohler dataset. Table 3 details the RMSE score. The RMSE score defines the absolute error between the desired and predicted scores. Henceforth, the lower the RMSE, the better the model is. Pearson correlation measures the comprehensive correspondence between the assignment of desired scores and

[2]Here by negative feature, we mean the learnt feature that results in adverse effects.

Table 4: Overview comparison of results on Mohler dataset with former approaches

Model/Approach	Features	RMSE	Pearson correlation
BOW (Mohler et al., 2011)	SVMRank	1.042	0.480
	SVR	0.999	0.431
tf-idf (Mohler et al., 2011))	SVR	1.022	0.327
tf-idf (Sultan et al., 2016)	LR + SIM	**0.887**	**0.592**
Word2Vec (Metzler, 2019)	SOWE + Verb phrases	1.025	0.458
	SIM+Verb phrases	1.016	0.488
GloVe (Metzler, 2019)	SOWE + Verb phrases	1.036	0.425
	SIM+Verb phrases	1.002	0.509
FastText (Metzler, 2019)	SOWE + Verb phrases	1.023	0.465
	SIM+Verb phrases	0.956	0.537
ELMo	SIM	0.978	0.485
GPT	SIM	1.082	0.248
BERT	SIM	1.057	0.318
GPT-2	SIM	1.065	0.311

predicted scores. Therefore, the higher the ρ, the better the model is. Table 4 provides a comparison of our results with former approaches and models.

ELMo embeddings attained a Pearson correlation of 0.485 and a RMSE score of 0.978 with isotonic regression. The results on linear and ridge regression are 0.451 and 0.449 for Pearson correlation; 0.995 and 0.996 for RMSE score respectively. GPT embeddings resulted in Pearson correlation of 0.248, 0.222 and 0.217[3]. The results of RMSE on GPT embeddings are 1.082, 1.088 and 1.089.

BERT embeddings also performed better with isotonic regression with a ρ of 0.318 and RMSE of 1.057 compared to linear (ρ : 0.266; RMSE: 1.077) and non-linear (ρ : 0.267; RMSE: 1.075) regressions. GPT-2 has performed alike BERT with ρ values of 0.311, 0.274 and 0.269 and RMSE scores of 1.065, 1.078 and 1.079 respectively on three regression training models.

Table 4 compares our results with conventional embeddings of Word2Vec, GloVe and FastText. Also, the traditional approaches of Bag-of-Words (BOW) and tf-idf on Mohler dataset are compared. However, we only consider the models or approaches that do not use domain-specific training of the data, for legible comparison. Various features or algorithms used by the models are demonstrated in the *Features*[4] column of the Table 4. The Pearson correlation of 0.592 and RMSE of 0.887 illustrate that tf-idf approach considering the combination of length ratio and similarity features by (Sultan et al., 2016) outperformed other approaches.

[3]We always provide our results in the order of isotonic, linear and ridge regressions when three consecutive results are detailed.

[4]SVM- Support Vector Machine; SVR - Support Vector Regression; LR- Length Ratio between desired answer and student answer; SIM - Similarity score; SOWE - Sum of Word Embeddings

7 OBSERVATIONS

Our results illustrate that ELMo model outperformed on domain-specific ASAG compared to other transfer learning models. The reasons that ELMo may have worked better than the other transfer learning models on the domain-specific dataset is two-fold. Firstly, the assignment of different vectors for the same word in different contexts. This is helpful for the homonyms. Secondly, the availability of significant amount of domain data in the pre-trained corpus compared to the other transfer learning models. The pretrained data of BERT, GPT and GPT-2 models is extensive and the domain which we are testing is comparatively very smaller. This resulted in the similarity scores in the range of 10^{-5} to 10^{-1}.

Isotonic regression worked better than the linear and ridge regressions. This is because, the isotonic regression trains step-wise, a similar way of assigning grades to the students manually, unlike linear and non-linear regression methods. We also noted that linear and ridge regression results of transfer learning models' were almost similar. This may be due to the significant linear fit of the data and the negligible non-linearity between the grade and the trained feature.

Compared to the other former approaches, considering the several preprocessing steps followed by multiple feature extraction and training, ELMo embeddings demonstrate competing results, without any preprocessing or multiple feature training.

8 CONCLUSION

We evaluated the embeddings of four transfer learning models on Mohler dataset (domain-specific) on the task of ASAG. These transfer learning models (ELMo, GPT, BERT and GPT-2) were explained breifly with their pretraining procedures. Besides, we also elucidated the ASAG task's significance and it's applications. The sentence embeddings are created from all the selected four transfer learning models for all the desired and student answers of the dataset. The encoding of answers are related to the words in the answers, irrespective of their order. The cosine similarity feature was extracted for every student answer and desired answer. This feature was trained with all the three isotonic, linear and ridge regression methods.

ELMo outperformed other transfer learning models on the task with a best RMSE score of 0.978 and Pearson correlation of 0.485. With these results, ELMo competed with the conventional word embeddings, such as Word2Vec, GloVe and FastText, without any preprocessing or multiple feature training. ELMo performed comparatively better than other transfer learning models, BERT, GPT and GPT-2. These transfer learning models have exhibited poor results on the Mohler dataset compared to the conventional word embeddings. We also concluded that ELMo can achieve near to the state of the art results without any further training of domain-specific data or compelling preprocessing of data.

9 FUTURE WORK

Although we have achieved significant results with ELMo without any preprocessing or multiple feature extraction or training, there is much scope to extend this work further. Firstly, it is important to consider the question demoting and elimination of stop words as in (Mohler et al., 2011), (Sultan et al., 2016) and (Metzler, 2019). The word alignment procedure by (Sultan et al., 2016), can have a substantial effect on sentence embeddings as most of the insignificant words can be removed.

Moreover, it is also important to explore different methods of assigning sentence embeddings such as mean-pooling, max-pooling, other than the sum of the vectors. The use of explicit sentence embeddings such as universal sentence encoders(Cer et al., 2018) can be effective to create a similarity feature. From the obtained results, it is also promising to use transfer learning embeddings trained or finetuned on domain-specific data. It is also necessary to change partisan datasets such as Mohler for better experimentation and evaluation.

ACKNOWLEDGEMENTS

We thank Tim Metzler for providing the updated version of the Mohler dataset for experimentation.

REFERENCES

Bachman, L. F., Carr, N., Kamei, G., Kim, M., Pan, M. J., Salvador, C., and Sawaki, Y. (2002). A reliable approach to automatic assessment of short answer free responses. In *Proceedings of the 19th international conference on Computational linguistics-Volume 2*, pages 1–4. Association for Computational Linguistics.

Bailey, S. and Meurers, D. (2008). Diagnosing meaning errors in short answers to reading comprehension questions. In *Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications*, pages 107–115.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bukai, O., Pokorny, R., and Haynes, J. (2006). An automated short-free-text scoring system: development and assessment. In *Proceedings of the Twentieth Interservice/Industry Training, Simulation, and Education Conference*, pages 1–11.

Burstein, J., Wolff, S., and Lu, C. (1999). Using lexical semantic techniques to classify free-responses. In *Breadth and depth of semantic lexicons*, pages 227–244. Springer.

Callear, D. H., Jerrams-Smith, J., and Soh, V. (2001). Caa of short non-mcq answers.

Cer, D., Yang, Y., Kong, S.-y., Hua, N., Limtiaco, N., John, R. S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., et al. (2018). Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Gütl, C. (2007). e-examiner: towards a fully-automatic knowledge assessment tool applicable in adaptive e-learning systems. In *Proceedings of the Second International Conference on Interactive Mobile and Computer Aided Learning*, pages 1–10. Citeseer.

Hou, W.-J. and Tsao, J.-H. (2011). Automatic assessment of students'free-text answers with different levels. *International Journal on Artificial Intelligence Tools*, 20(02):327–347.

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., and Wu, Y. (2016). Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*.

Kim, Y., Jernite, Y., Sontag, D., and Rush, A. M. (2016). Character-aware neural language models. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Kumar, R. (2018). Evaluation of semantic textual similarity approaches for automatic short answer grading.

Metzler, T. D. (2019). Computer-assisted grading of short answers using word embeddings and keyphrase extraction.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Mitchell, T., Russell, T., Broomhead, P., and Aldridge, N. (2002). Towards robust computerised marking of free-text responses.

Mohler, M., Bunescu, R., and Mihalcea, R. (2011). Learning to grade short answer questions using semantic similarity measures and dependency graph alignments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 752–762. Association for Computational Linguistics.

Mohler, M. and Mihalcea, R. (2009). 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.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Pérez, D., Alfonseca, E., Rodríguez, P., Gliozzo, A., Strapparava, C., and Magnini, B. (2005). About the effects of combining latent semantic analysis with natural language processing techniques for free-text assessment. *Revista signos*, 38(59):325–343.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. (2018). Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Rocktäschel, T., Grefenstette, E., Hermann, K. M., Kočiskỳ, T., and Blunsom, P. (2015). Reasoning about entailment with neural attention. *arXiv preprint arXiv:1509.06664*.

Sultan, M. A., Salazar, C., and Sumner, T. (2016). 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*, pages 1070–1075.

Thomas, P. (2003). The evaluation of electronic marking of examinations. In *Proceedings of the 8th annual conference on Innovation and technology in computer science education*, pages 50–54.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Wang, H.-C., Chang, C.-Y., and Li, T.-Y. (2008). Assessing creative problem-solving with automated text grading. *Computers & Education*, 51(4):1450–1466.

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*.