A Meta-Learning Augmented Bidirectional Transformer Model for Automatic Short Answer Grading

Zichao Wang
Rice University
Houston, TX 77005
jzwang@rice.edu

Andrew S. Lan
University of Massachusetts Amherst
Amherst, MA 01003
andrewlan@cs.umass.edu

Andrew E. Waters
OpenStax
Houston, TX 77005
aew2@rice.edu

Phillip Grimaldi
OpenStax
Houston, TX 77005
phillip.grimaldi@rice.edu

Richard G. Baraniuk
Rice University & OpenStax
Houston, TX 77005
richb@rice.edu

ABSTRACT

We introduce ml-BERT, an effective machine learning method for automatic short answer grading when training data, i.e., graded answers, is limited. Our method combines BERT (Bidirectional Representation of the Transformer), the state-of-the-art model for learning textual data representations, with meta-learning, a training framework that leverages additional data and learning tasks to improve model performance when labeled data is limited. Our intuition is to use meta-learning to help us learn an *initialization* of the BERT parameters in a specific target subject domain using unlabeled data, thus fully leveraging the limited labeled training data for the grading task. Experiments on a real-world student answer dataset demonstrate the promise of ml-BERT method for automatic short answer grading.

1. INTRODUCTION

We consider the problem of automatic grading for short answer questions that require students to provide concise *textual* responses. Figure 1 shows an example of short answer questions. The pedagogical benefits of these questions over multiple-choice questions have been studied and validated by [11]. Unfortunately, manually grading short answer questions is labor intensive, rendering it extremely challenging to administer these questions in large-scale educational settings.

In this paper, we aim to advance the state-of-the-art in automatic short answer grading. This problem can be viewed as a supervised (machine) learning problem where we would like to grade (classify) answers as correct or incorrect given a dataset of answers and their grades (labels).[1] A method capable of reliably grading short answer questions gives instructors more freedom in choosing the form of assessments in large-scale educational settings where manual grading is unfeasible. In practice, however, the number of graded answers is very limited since manual grading is labor-intensive. This constraint makes training a classifier for automatic grading a highly challenging task. Thus, it is desirable to develop a method that can effectively leverage the ungraded answers to learn a representation of the answers and achieve satisfactory performance with only a few graded answers. This feature is valuable for instructors in large-scale educational settings because they would need to manually grade only a few answers, which saves them time and effort that are better allocated to other pedagogical actions. Therefore, in this work, we focus on this particular scenario where we only have access to a limited number of graded answers.

Question:	What is a difference between saturated and unsaturated fats?
Answer:	Saturated fats have no double bonds whereas unsaturated fats have at least one double bond

Figure 1: A sample question and correct student answer in their raw textual form taken from a high school biology class.

A number of prior works have demonstrated the effectiveness of machine learning methods in automatic grading [2, 8, 9, 10]. As a high-level summary, a typical method first converts textual data composed of questions and answers to some vector representations and then uses them to classify an answer as correct or incorrect. In our problem setting, however, this method for automatic short answer grading faces two major challenges. First, we need a *model* capable of producing high quality representations of textual data that captures the information in both the question text and the answer text, which is essential for the classifier to make its decision. We note that learning a representation of textual data has been a challenge not only for short answer grading but also for the broader field of natural language processing (NLP). Second, since the number of graded answers is limited, we need a specialized *training procedure* that enables the classifier to remain effective using only a limited number of labels. We note that this procedure is important because a number of prior works, e.g., [4], have demonstrated that regular training procedures may result in poor results when labeled data is limited. For example, we have merely less than 20k labeled answers whereas related classification tasks such as sentiment analysis have over 1 million labeled examples [5], which contributes to the success of machine learning models on those tasks.

[1] We will use “grade” and “label” interchangeably throughout the paper.

Zichao Wang, Andrew Lan, Andrew Waters, Phillip Grimaldi and Richard Baraniuk "A Meta-Learning Approach to Automatic Short Answer Grading" In: *Proceedings of The 12th International Conference on Educational Data Mining (EDM 2019),* Collin F. Lynch, Agathe Merceron, Michel Desmarais, & Roger Nkambou (eds.) 2019, pp. 667 - 670

To tackle the first challenge and learn textual data representations efficiently and effectively, we resort to BERT (bidirectional encoder representation of the transformer), the state-of-the-art model for text. Since BERT achieves the best results to date in a wide range of benchmark NLP tasks, we develop our method on top of it. To tackle the second challenge and fight the scarcity of labeled data, we resort to *meta-learning*, an emerging training procedure that searches for a good initialization of the model parameters by using unlabeled data to optimize additional objectives that are unrelated to the target task. Previous studies [4, 7] have shown that parameter initialization is critical for the success of neural network models including BERT; Thus, the meta-learning procedure helps us to find a better initialization of the model parameters and quickly adapt to new tasks with only limited labeled data.

1.1 Contributions

We propose ml-BERT, a meta-learning method that augments the bidirectional encoder representation of the transformer (BERT) model for automatic short answer grading with limited labeled answers. Our method consists of two learning phases. First, in the meta-learning phase, we optimize for an initialization of BERT parameters using unlabeled data. We carefully choose data from a specific educational domain to use in the meta-learning phase such that it is closely related to the questions in the short answer grading task. This choice ensures that the meta-learned initialization builds a good representation of the domain and is highly useful to the subsequent short answer grading task. Second, in the regular learning phase, we optimize BERT parameters for the short answer grading task using limited labeled training data, starting from the meta-learned parameter initialization in the previous phase. Experimental results on a real-world dataset consisting of student answers to a set of questions in high school biology shows that the proposed ml-BERT method is more effective than regular BERT and several other baseline methods when labeled data is limited.

The rest of the paper is organized as follows. Section 2 details our ml-BERT method. Section 3 presents experimental results on a real-world student answer dataset and discusses the key findings. Finally, Section 4 summarizes our work and proposes future research directions. This paper is a truncated version; for more details including background on BERT and meta-learning, exhaustive literature review and additional experimental results, please refer to the long version of this paper on ArXiv with the same title.

2. METHOD

In this section, we describe ml-BERT, our novel training procedure that augments BERT with meta-learning for short answer grading. The ml-BERT training procedure consists of two learning phases, namely, the *meta-learning phase* and the *regular learning phase*. In the meta-learning phase, we optimize for an initialization of BERT parameters that ultimately leads to better short answer grading performance. In the regular learning phase, we optimize for the BERT parameters using labeled short answer dataset to further improve the model performance on short answer grading.

Key to the success of ml-BERT is the choice of datasets and tasks in the meta-learning phase for learning parameter initialization. We choose *language modeling* and *next sentence prediction* as our meta-learning tasks. We use a dataset that includes textbooks, question texts, and correct student answer texts for the language modeling task and a dataset that includes textbooks for the next sentence prediction task. The specific choice of the above data depends on the educational domain. For example, we would choose relevant biology textbooks if the task is to grade answers to biology related questions. Intuitively, the language modeling task helps us learn an initialization that captures the nuances of the language usage specific to each educational domain. The next sentence prediction task helps us learn an initialization that captures the logic flow underlying the textbooks from which questions are asked. Initialization learned from the above tasks and data is thus informative to the subsequent short answer grading task because it contains some knowledge of whether an answer uses appropriate language and whether an answer is logically related to its associated question and the specific educational domain.

Below, we first formally define the target task which is short answer grading. We then describe the two meta-learning tasks and their datasets. Finally, we present the full ml-BERT training procedure.

Target task: short answer grading. This is a supervised learning problem that we explicitly train the model to perform well on. The training dataset for short answer grading consists of N examples $\{q_i, a_i, y_i\}_i^N$, where q_i and a_i are text segments representing question and student short answer, respectively, and y_i is a binary variable indicating whether the answer is correct or incorrect. The learning objective is to correctly classify each answer given the question. We use the negative log-likelihood loss to measure this objective:

$$\mathcal{L}_T = -\frac{1}{N}\sum_{i=1}^{N} \log p(y_i|a_i, q_i) \tag{1}$$

where $p(y_i|a_i, q_i)$ is modeled by a composition of functions including BERT mapping $f^{(1)}(\cdot)$ of the first "`[CLS]`" token (see Section 2.1 of the long version of this paper),[2] a fully connected layer $g : \mathbb{R}^{\mathbb{M}} \to \mathbb{R}$, and a sigmoid function $\sigma(\cdot)$:

$$p(y_i|a_i, q_i) = \sigma\left(g\left(f^{(1)}\left(a_i, q_i\right)\right)\right).$$

In practice, in order to obtain a single textual input for BERT, we append all tokens of each answer a_i to all tokens of its associated question q_i, append "`[CLS]`" and `[SEP]`" respectively at the beginning and end of the concatenated token sequence, and separate the answer and question tokens with "`[SEP]`".

Meta-learning task #1: language modeling. This is an unsupervised learning problem. The dataset $\mathcal{D}_1 = \{s_i\}_{i=1}^{N_1}$ consists of a collection of N_1 sentences s_i taken from text corpora specific to the educational subject. Each sentence is represented by a sequence of J_i tokens: $s_i = \{w_j\}_{j=1}^{J_i}$ (see Section 2.1 of the long version of this paper).

During training, we randomly replace a portion of all tokens in each sentence with the special token "`[MASK]`" and ask the model to predict these masked tokens. We set the portion of masked tokens to 15% for each sentence. Performance of token prediction is measured by negative log-likelihood loss $\mathcal{L}_1$ as:

$$\mathcal{L}_1 = -\frac{1}{N_1}\sum_{i=1}^{N_1}\sum_{w_{ik}} \log p(w_{ik}|w_1, \ldots, w_{ik-1}, w_{ik+1}, \ldots, w_{J_i}) \tag{2}$$

where w_{ik} represents the k^{th} masked token in the i^{th} sentence. The conditional probability distribution is modeled using the composition of BERT mapping $f(\cdot)$, a linear layer $g_i : \mathbb{R}^M \to \mathbb{R}^V$

[2] In BERT, we obtain the sentence representation by quering the embedding of this "`[CLS]`" token.

Table 1: Statistics of the dataset for each task. Note that the short answer grading dataset consists of only labeled answers which represents less than 10% of all available answers.

tasks	#examples	#average tokens
Short answer grading	495 questions 18143 answers	17.88 per question 12.31 per answer
Masked language modeling	78684 sentences	19.08 per sentence
Next sentence prediction	21052 sentences	25.54 per sentence

that maps every BERT encoded token into a vector of dimension V which is the size of the vocabulary, and a softmax function:

$$p(w_k|w_1, \cdots, w_{k-1}, w_{k+1}, \cdots, w_J) = \text{softmax}\left(g_1(f(\tilde{s}))\right)$$

where we drop the example index i for notation simplicity. $\tilde{s}$ denotes the input sentence with the k^{th} token replaced by the mask which indicates which token the model should predict.

We note that there are other strategies for language modeling with BERT. Conventionally, one models language by predicting the next token in the input sentence given all previous tokens [6]. We choose this *masked* prediction strategy for language modeling because of its superior empirical performance than the conventional approach [3].

Meta-learning Task #2: next sentence prediction. This is an unsupervised learning problem. The dataset $\mathcal{D}_2 = \{s_i, \hat{s}_i, z_i\}_{i=1}^{N_2}$ consists of N_2 pairs of sentences sampled from textbook chapters specific to the educational subject of the short-answer questions. We sample the sentence $\hat{s}_i$ such that half of the time it is the next sentence of s_i and half of the time it is a sentence from a random location in the textbook. z_i is a binary variable indicating whether $\hat{s}_i$ is the next sentence of s_i. We use negative log likelihood $\mathcal{L}_2$ to measure model performance on next sentence prediction:

$$\mathcal{L}_2 = -\frac{1}{N_2}\sum_{i=1}^{N_2} \log p(z_i|s_i, \hat{s}_i) \tag{3}$$

where the conditional probability is modeled by the composition of BERT mapping $f^{(1)}(\cdot)$ of the first "`[CLS]`" token, a fully connected layer $g_2 : \mathbb{R}^M \to \mathbb{R}$, and a sigmoid function $\sigma(\cdot)$:

$$p(y_i|s_i, \hat{s}_i) = \sigma\left(g_2\left(f^{(1)}\left(s_i, \hat{s}_i\right)\right)\right) .$$

The ml-BERT training procedure. Model parameter update according to ml-BERT proceeds as follows. In the meta-learning phase, we update the BERT parameters by alternating between the two tasks until we reach a pre-specified stopping condition. In the regular learning phase, we initialize the BERT parameters from the meta-learned parameter initializations in the previous phase and update the parameters using the labeled short answer dataset until we reach a pre-specified stopping condition. In both phases, we perform parameter update using gradient descent optimizers. The exact choice of the optimizer is flexible; we use the customized Adam optimizer for BERT outlined in [3]. The proposed ml-BERT procedure is simple to implement and effective in practice, which we demonstrate in the next section.

3. EXPERIMENTAL RESULTS

We now demonstrate the effectiveness of our method using real-world data. We first introduce various model and training settings and then explain our results in detail. Code for our experiments can be shared upon request.

Dataset and pre-processing steps. We collect real-world short answers from semester-long biology classes that use the OpenStax Biology textbook.[3] Table 1 summarizes the key statistics of the dataset. Notably, only about 10% of the answers are graded; we use only the graded responses as training data. Heaping the power of the vast number of ungraded responses is left for future work. We perform an 80/20 training/validation split independently for each question. We also discard questions in the training set with less than 20 labeled answers, 10 for each label, and questions in the validation set with less than 4 labeled answers, 2 for each label. This is to ensure we have well-balanced training and validation splits of the dataset. For the language modeling task during meta-learning, we use all textbook text, questions, instructor-authored answers associated with the OpenStax Biology textbook. For the next sentence prediction task, we only use the textbook text.

The textual data is minimally pre-processed. We first turn all texts to lowercase. For BERT, we use the WordPiece tokenizer to process the input text into a sequence of tokens. For other baseline models, we use a bag-of-words (BoW) representation of textual data, which involves a more complicated pre-processing pipeline including tokenization, lemmatization and stop word removal.

Hyper-parameters. We adapt the standard BERT model configuration and training setup. We use BERT's customized Adam optimizer with a learning rate of 3×10^{-5}. Training lasts 10 epochs with a batch size of 32. Each question-answer pair is limited to 128 word pieces; shorter ones are padded and longer ones are truncated. Most of the setup information is available in the official BERT code release.[4]

Baseline models. We compare ml-BERT with four baseline methods: 1) baseline BERT without meta-learning, 2) Random Forest, 3) K-Nearest Neighbors and 4) logistic regression. The baseline BERT model has the same hyper-parameter setup as BERT with meta-learning. We use default settings in the python `sklearn` package for the remaining three models. In addition, logistic regression is the best linear classifier among those that we tested; therefore we use logistic regression as the representative linear classifier baseline.[5]

3.1 Quantitative Analysis

Table 2 compare the classification accuracy and F1 score of ml-BERT to that of the four baseline models on the validation set. We clearly see ml-BERT improves both classification accuracy and F1 score compared to the baseline models. Note in particular that BERT without meta-learning only achieves comparable performance with random forest, but with meta-learning it is able to outperform all baseline models. This suggest that in the limited labeled answer scenario, meta-learning has the potential to lift the model performance further.

To get a fine-grained performance analysis, We compare the performance of ml-BERT and BERT on questions of different levels according to Bloom Taxonomy. Bloom Taxonomy [1] categorizes

[3] https://openstax.org/details/books/biology

[4] https://github.com/google-research/bert

[5] Recall that K-Nearest Neighbors and Random Forest are nonlinear classifiers.

Table 2: Evaluation accuracy and F1 score comparing ml-BERT with various baselines.

Models	eval. acc.	eval. F1
Logistic Regression	71.39%	0.723
K Nearest Neighbors	72.74%	0.735
Random Forest	77.82%	0.768
Baseline BERT	77.80%	0.788
ml-BERT	**80.17%**	**0.815**

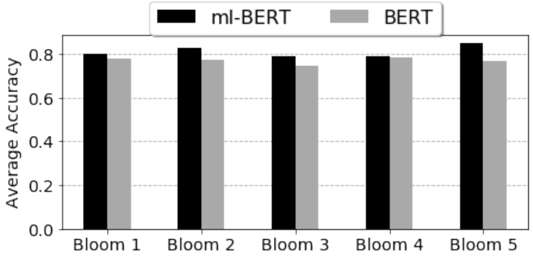

Figure 2: Grading accuracy per Bloom level comparing ml-BERT to BERT. We see improved accuracy across questions with different Bloom's levels.

questions into 6 distinct levels where higher level indicates more cognitive load required to answer a question. We can thus use Bloom levels as rough difficulty measure of the questions. Figure 2 summarizes the classification accuracy for each bloom level. We see that ml-BERT outperforms BERT in grading questions across all bloom levels, most significantly at Bloom level 5. This can be explained by the benefit of meta-learning which learns language patterns and logical relations in the specific educational subject that helps grade these "difficult" questions with higher accuracy.

3.2 Influence of Each Meta-Learning Task

We investigate the effect of each of the meta-learning tasks on the quality of the learned representations of text. Specifically, we compare ml-BERT using both learning tasks with ml-BERT using only one of the learning tasks. We summarize the comparison in Table 3. Results suggest that, overall, both tasks contribute to the quality of the learned representations. We emphasize that, even with only one of the meta-learning tasks, ml-BERT is able to adapt to the short answer grading task and achieve better performance compared to baseline BERT. This observation highlights the the importance of using task-specific data and tasks in the meta-learning phase. Moreover, it also suggests that ml-BERT has potential in further improving its performance as we incorporate more tasks specific to the target educational domain, which we leave for future work.

4. CONCLUSIONS AND FUTURE WORK

We have introduced ml-BERT, a method to augment the state-of-the-art text embedding model BERT with meta-learning to improve its performance on automatic short answer grading. In the first phase of ml-BERT, we use meta-learning to learn an initialization of BERT parameters by training language modeling and next sentence prediction tasks on data specific to the relevant educational domain. In the second phase, we further optimize the model parameters using limited labels on the correctness of the short answers. We experimentally validate the effectiveness of ml-BERT on real-world student answers collected in high school biology classes. Both quantitative and quantitative results demonstrate that the proposed method is promising.

Table 3: The impact of each of the two meta-learning tasks on grading accuracy. We see that the best results are achieved when we use both meta-learning tasks. We also observe that using either one of the meta-learning tasks already leads to improvement over baseline BERT.

Conditions	eval. acc.	eval. F1
ml-BERT		
- Without masked language modeling	79.12%	0.805
- Without next sentence prediction	78.63%	0.793

5. ACKNOWLEDGEMENTS

We thank the anonymous reviewers for their constructive feedback and Shashank Sonkar for helpful discussions. This work was partially supported by Bill and Melinda Gates Foundation, Arthur & Carlyse Ciocca Charitable Foundation, NSF grants IIS-17-30574 and IIS-18-38177, and DOD Vannevar Bush Faculty Fellowship (NSSEFF) grant N00014-18-1-2047.

6. REFERENCES

[1] P. Armstrong. *Bloom's Taxonomy*, 2014 (accessed Mar. 4, 2019).

[2] S. Burrows, I. Gurevych, and B. Stein. The eras and trends of automatic short answer grading. *Int. J. Artificial Intell. in Edu.*, 25(1):60–117, Mar. 2015.

[3] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv e-prints*, 1810.04805, Oct 2018.

[4] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta learning for fast adaptation of deep networks. In *Proc. Int. Conf. Mach. Learn.*, volume 70, pages 1126–1135, Aug. 2017.

[5] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *CS224N Project Report*, 2009.

[6] D. Jurafsky and J. H. Martin. *Speech and Language Processing (2nd Edition)*. Prentice-Hall, Inc., 2009.

[7] Q. V. Le, N. Jaitly, and G. E. Hinton. A Simple Way to Initialize Recurrent Networks of Rectified Linear Units. *ArXiv e-prints*, 1504.00941, Apr. 2015.

[8] M. Mohler and R. Mihalcea. Text-to-text semantic similarity for automatic short answer grading. In *Proc. Conf. Eur. Chapter Assoc. Comput. Linguistics*, pages 567–575, Mar. 2009.

[9] M. A. Sultan, C. Salazar, and T. Sumner. Fast and easy short answer grading with high accuracy. In *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Human Language Technol.*, pages 1070–1075, Jun. 2016.

[10] N. Suzen, A. Gorban, J. Levesley, and E. Mirkes. Automatic Short Answer Grading and Feedback Using Text Mining Methods. *arXiv e-prints*, 1807.10543, Jul. 2018.

[11] A. Waters, P. Grimaldi, A. S. Lan, and R. G. Baraniuk. Short-answer responses to stem exercises: Measuring response validity and its impact on learning. In *Proc. Conf. Edu. Data Mining*, pages 374–375, Jun. 2017.