A Meta-Learning Augmented Bidirectional Transformer Model for Automatic Short Answer Grading
Zichao (Jack) Wang¹, Andrew S. Lan³, Andrew Waters², Phillip Grimaldi², Richard G. Baraniuk¹,²

1. Department of Electrical and Computer Engineering, Rice University; 2. OpenStax; 3. College of Information and Computer Sciences, University of Massachusetts Amherst

Motivation
- Design automatic grading methods for textual responses to aid grading assignments that require textual responses in large scale educational scenarios
- Treat as a binary classification problem (correct / incorrect)
- Challenge: data-driven methods rely on massive labeled data but may be limited in quantity
- Idea: use meta learning in this limited training data scenario

Background: Meta Learning
- Learn any hyper-parameter of the experiment, including
  - Parameter initialization
  - Learning rate
  - Batch size
- Learn with generally available data sets and tasks

The ml-BERT Method
- Meta-learn an initialization of the BERT parameters
- Meta-learn on two tasks
  - Masked language modeling
  - Next sentence prediction
- Meta-learn on educational data sets
  - Textbooks
  - Question texts
  - Correct students' textual responses
- After meta-learning, fine-tune on labeled students' graded textual responses (treat as binary classification task)

Input: \( \mathcal{D}, \mathcal{L}, \mathcal{D}_1, \mathcal{L}_1, \mathcal{D}_2, \mathcal{L}_2 \) which are the datasets and learning objectives of the target task and the two learning tasks
Output: \( \Theta \) which is the learned BERT parameters

```
while not done do
    for batch of data in \( \mathcal{D}_1 \) do
        Evaluate \( \mathcal{L}_1 \) using the batch of data according to (2)
        Evaluate \( \nabla_{\theta_1} \mathcal{L}_1 \) and update \( \theta_1 \) for each \( \theta_1 \in \Theta \)
    end
    for batch of data in \( \mathcal{D}_2 \) do
        Evaluate \( \mathcal{L}_2 \) using the batch of data on the mask language modeling task
        Evaluate \( \nabla_{\theta_2} \mathcal{L}_2 \) and update \( \theta_2 \) for each \( \theta_2 \in \Theta \)
    end
    end
while not done do
    for batch of data in \( \mathcal{D} \) do
        Evaluate \( \mathcal{L} \) using the batch of data on the next sentence prediction task
        Evaluate \( \nabla_{\theta_1} \mathcal{L} \) and update \( \theta_1 \) for each \( \theta_1 \in \Theta \)
    end
```

Experimental Setup
- We evaluate on students' textual responses from OpenStax Biology textbook. Dataset statistics below.

<table>
<thead>
<tr>
<th>tasks</th>
<th>#examples</th>
<th>#average tokens</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short answer grading</td>
<td>495</td>
<td>17.88 per question</td>
</tr>
<tr>
<td>Masked language modeling</td>
<td>78684</td>
<td>19.08 per sentence</td>
</tr>
<tr>
<td>Next sentence prediction</td>
<td>21052</td>
<td>25.54 per sentence</td>
</tr>
</tbody>
</table>

Experimental Results
- Evaluation accuracy and F1 score comparing ml-BERT with various baselines.

<table>
<thead>
<tr>
<th>Models</th>
<th>eval. acc.</th>
<th>eval. F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>71.39%</td>
<td>0.723</td>
</tr>
<tr>
<td>K Nearest Neighbors</td>
<td>72.74%</td>
<td>0.735</td>
</tr>
<tr>
<td>Random Forest</td>
<td>77.82%</td>
<td>0.768</td>
</tr>
<tr>
<td>Baseline BERT</td>
<td>77.80%</td>
<td>0.788</td>
</tr>
<tr>
<td>ml-BERT</td>
<td>80.17%</td>
<td>0.815</td>
</tr>
</tbody>
</table>

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

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.

Contact: Zichao (Jack) Wang zw16@rice.edu