Automatic Short Answer Grading & Autograding with RoBERTa

Yehchan Yoo

Introducing Myself!

- Currently a UC Berkeley undergraduate majoring in Statistics and Political Economy (& minoring in Data Science)
- Began as a research apprentice through URAP during Spring 2023; currently a part-time research assistant
- Part of the DDM team (though the project itself is part of PSM)

General Goals

The autograder is intended to address the following two issues in scoring responses to constructed-response (CR) items:

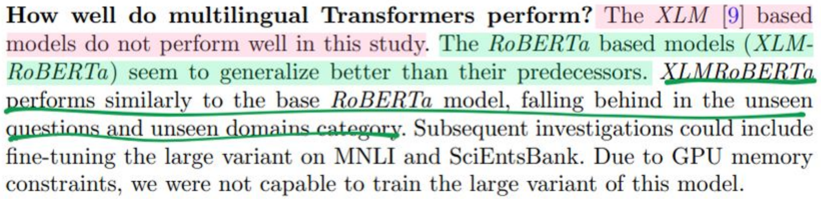
1. Time burden on human scorers
2. Rater reliability (and bias)

Our goal is not to replace human raters with the autograder. Rather, we intend the autograder to support human raters.

Additionally, in the long run,

3. Provide automated feedback to students

Background



How well do multilingual Transformers perform? The *XLM* [9] based models do not perform well in this study. The *RoBERTa* based models (*XLM-RoBERTa*) seem to generalize better than their predecessors. *XLMRoBERTa* performs similarly to the base *RoBERTa* model, falling behind in the unseen questions and unseen domains category. Subsequent investigations could include fine-tuning the large variant on MNLI and SciEntsBank. Due to GPU memory constraints, we were not capable to train the large variant of this model.

(Camus and Filighera, 2020, p. 46)

- Joined the project late – around July 2023
- Use of RoBERTa model inspired by paper by Camus and Filighera, as first introduced by Ms. Aubrey Condor
- Inspired to make another model (in addition to Will's model) to see how RoBERTa would perform in comparison to Sentence-BERT (Will's model as of July 2023)

RoBERTa - A **R**obustly **O**ptimized **BERT** Pretraining **A**pproach

- Among several BERT models tested for ASAG in a research paper, RoBERTa model found to be the best for generalization (Camus and Filighera, 2020, p. 46)
- BERT – a transformer language model introduced by Google in 2018
- RoBERTa - a more optimized version of BERT, developed by Meta AI in 2019
  - Developed in PyTorch; achieves better optimization via removal of next-sentence pretraining goal, much bigger mini-batches, much greater learning rates, much more training data, et cetera (*RoBERTa: An Optimized Method for Pretraining Self-Supervised NLP Systems*, 2019)
    - **Next-sentence pretraining goal removed to improve downstream task performance**, i.e. performance in tasks such as figuring out relationships between sentence pairs (Liu et al, 2019, p. 5)

Data Used

- TreeGrowth.03ab_MCOE (Fall 2022 PSM)
  - Most recent data from AG Pilot datasets
  - Dataset well-organized in spreadsheets
  - Output consists of an integer score from 1-3
    - Score changed from 1-3 to 0-2 for reducing technical issues

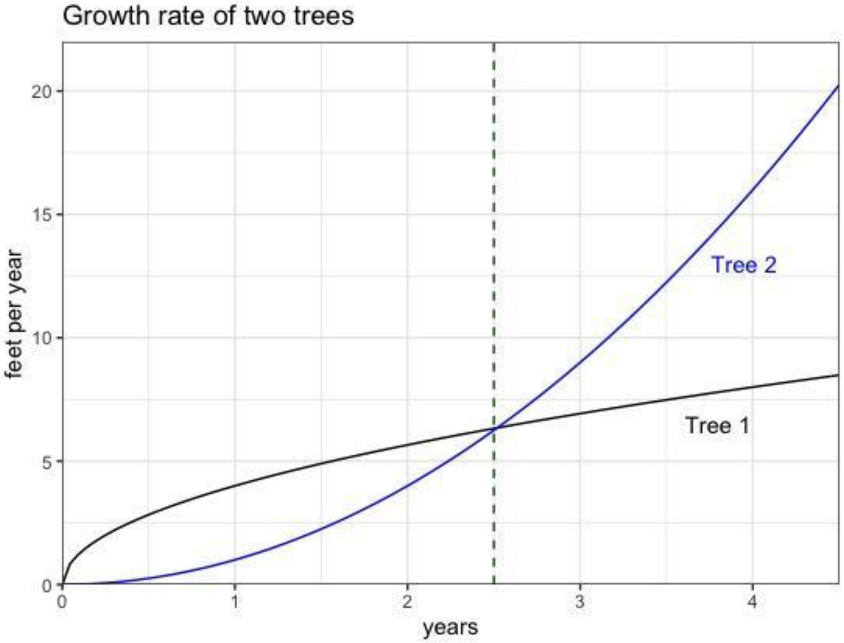


Image from TreeGrowth.03ab_MCOE

Dealing with Ordinal Nature of Grades

- Rubric scores constitute an ordinal variable, i.e. a categorical variable with a natural order
  - Must take into account not only the ordering of the variable categories, but also the categorical nature of the variable (meaning that we can have scores of 1, 2, 3, etc., but not 1.5, 2.5, etc.)
- Also must find a way to aggregate multiple grades from multiple raters
- Solution:
  - Used **median grade** as the standard grade for each answer due to optimality property
  - Handled non-integer medians by randomly rounding those medians up or down to the nearest integer with equal probability

General Procedure

1. Extract the input text explanations and the output scores from the original dataset.
2. Start up a pre-trained RoBERTa model
   a. Use of pre-trained model allows for an accurate autograder model without having too much training data at hand
3. Further train the model to fit the training data

Prototype

- Developed on Google Colab
  - Took advantage of Google's Nvidia T4 GPU setup for significantly improved performance
- Training and test code made with help of LLMs (Claude 2, Bard, ChatGPT)
- All files and work uploaded on Github and being tracked with Git (on a private repository)
- Made one model for each dataset with 80-10-10 train-validation-test split
  - Training solely done on the scores and the textual answers, as scores with wrong multiple choice answers were already removed from the dataset

Result

- Achieved a fairly high ~75% test accuracy (as of October 9, 2023) over the TG dataset!
  - Initially achieved 52% test accuracy when the model was proposed on August 2023
  - Test accuracy improved through experimentation with hyperparameters (mainly learning rate) with validation set + increase in number of epochs (from 5 to 10)
- Still running into technical difficulties, mainly with performance
  - Hyperparameter tuning very difficult due to lack of provided GPU memory
    - One trained model took up ~7 GB of GPU memory – around half of what was provided
  - Tried to use DistilRobertaModel instead of the full RoBERTa model, which helped but was not enough
  - Running multiple models in one Colab notebook also difficult due to issues with CUDA – further hampering hyperparameter tuning and making it difficult to work with multiple datasets at once

Moving Forward

- Can try to use metrics other than simple accuracy (e.g. F1 score) to gauge model accuracy
- Classification for subgrades (e.g. 1A, 1B, 1C)
    - Subgrades for a certain score have the same ordering and act as a nominal (not ordinal) variable
    - Means that scores act *partially* as a nominal variable (with subgrades) and *partially* as an ordinal variable, leading to high complexity and potentially a need for multiple models
    - Considering making one model just for ordinal part of the grading rubric (1, 2, 3, etc…) and then another model for further classifying the grades into subgrades (1A, 1B, 1C, etc…)
        - But…this can lead to performance issues, as noted previously
- May also consider using ensemble methods (e.g. voting) with other models (e.g. random forest) to improve performance
    - But this also comes with a risk of worsening performance issues
- Interpretability
    - Current classification model performs well, but does not tell how it got its predictions
    - Allowing a much more complicated grading scheme (e.g. Dedoose scheme) with multiple subcategories can increase interpretability, but requires significant amount of graphics performance
    - Can do inference on the model later on to see what phrases match up with answers from certain score categories
- Generalizability of the model (for different types of rubrics)
- Maybe use different open-source language models? (Llama, ChatGPT2, etc.)

Related literature

keywords:

automatic short answer grading(ASAG)

automated essay scoring (AES)

Ahmed, A., Joorabchi, A., & Hayes, M. J. (2022). On Deep Learning Approaches to Automated Assessment: Strategies for Short Answer Grading. *CSEDU (2)*, 85-94. **[PDF]** scitepress.org

Basu, S., Jacobs, C., & Vanderwende, L. (2013). Powergrading: a clustering approach to amplify human effort for short answer grading. *Transactions of the Association for Computational Linguistics*, *1*, 391-402. **[PDF]** mit.edu

Bonthu, S., Sree, S. R., & Prasad, M. K. (2023). Improving the performance of automatic short answer grading using transfer learning and augmentation. *Engineering Applications of Artificial Intelligence*, *123*, 106292. **[HTML]** sciencedirect.com

*Burrows, S., Gurevych, I., & Stein, B. (2015). The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, *25*, 60-117. **[HTML]** springer.com

*Camus, L., & Filighera, A. (2020). Investigating transformers for automatic short answer grading. In *Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part II 21* (pp. 43-48). Springer International Publishing. **[HTML]** nih.gov

Gaddipati, S. K., Nair, D., & Plöger, P. G. (2020). Comparative evaluation of pretrained transfer learning models on automatic short answer grading. *arXiv preprint arXiv:2009.01303*. **[PDF]** arxiv.org

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

Liu, T., Ding, W., Wang, Z., Tang, J., Huang, G. Y., & Liu, Z. (2019). Automatic short answer grading via multiway attention networks. In *Artificial Intelligence in Education: 20th International Conference, AIED 2019, Chicago, IL, USA, June 25-29, 2019, Proceedings, Part II 20* (pp. 169-173). Springer International Publishing. **[PDF]** arxiv.org

*Sung, C., Dhamecha, T., Saha, S., Ma, T., Reddy, V., & Arora, R. (2019, November). Pre-training BERT on domain resources for short answer grading. 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. 6071-6075).

Uto, M., & Okano, M. (2020). Robust neural automated essay scoring using item response theory. In *Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part I 21* (pp. 549-561). Springer International Publishing. **[HTML]** nih.gov

*Galhardi, L. B., & Brancher, J. D. (2018). Machine learning approach for automatic short answer grading: A systematic review. In *Advances in Artificial Intelligence-IBERAMIA 2018: 16th Ibero-American Conference on AI, Trujillo, Peru, November 13-16, 2018, Proceedings 16* (pp. 380-391). Springer International Publishing.

Thank you!