The Grades of Wrath: A Two-Stage BERT Model for AES

Collin Chee
UC Berkeley School of Information
cmchee@berkeley.edu

Cameron Wright
UC Berkeley School of Information
cbwright@berkeley.edu

Abstract

We propose a two-stage model for Automated Essay Scoring (AES): one stage which extracts relevant features from an essay, and another which uses those features to derive an overall essay score. The two stages are trained on two separate datasets, the first of which provides 6 scores for each essay (cohesion, syntax, vocabulary, phraseology, grammar, and conventions), the second of which provides just an overall essay score. We evaluate our model using Quadratic Weighted Kappa (QWK) as it is the reigning evaluation metric used in the field of AES. Several versions of each stage are explored and we report the final test QWK for each one. The version of our model that performed the best earned a test QWK score of **0.441** and is structured as follows: (stage 1) a BERT base model followed by a feed-forward neural network with 6 regression heads, (stage 2) a feed-forward neural network. Our work adds novelty to the AES field by offering a prompt independent approach to essay grading. Further research could explore methods of receiving the essay prompt as an input to the grading system.

1 Introduction

Scoring essays on standardized tests can be extremely time consuming and laborious for human graders. The grading process is completely different from standard multiple choice answers as the evaluation of an essay is a subjective topic. AES has always been a challenging task to implement into standardized test grading as models must be able to capture different aspects of essays such as grammar and cohesion. This research aims to create an AES model that will utilize NLP techniques to accurately assess different dimensions of an essay and score them.

A high quality AES system would have the benefits of reducing time, cost, and bias in the essay grading process. Zupanc and Bosnic (2018) state that subjectivity is evident in human grading given that multiple graders of the same essay often award different scores. Eliminating bias was the main appeal of this project for us, but unfortunately our datasets did not provide demographic information about the students so we were unable to address this goal. The features that contribute to the quality of an essay can be complex and challenging for an autonomous system to identify. A realistic danger is that a model might learn to associate simpler - but undeserving - essay characteristics with the essay score instead.

This research aims to transfer models from two different datasets to explore the possibility of a general automated essay grader.

2 Background

Previous work on Hewlett Foundation's Kaggle competition (Hamner et al., 2011) has illustrated many different methods of creating an automated grading system. Since the competition was released in 2011, the initial stages of architectures consisted of a combination of LSTM and CNN models. Alikaniotis et al. (2016) utilized a corpus model with pre-trained embeddings as a way of capturing how each word contributes towards the final score of an essay. Combined with an LSTM that is able to understand the context of sentences, the researchers were able to generate strong essay score predictions. The researchers may have generated extremely good results, but they have commented that their methods are for specifically this dataset.

Beyond 2017 saw many different research papers using transformers, especially BERT, to model the scoring process. Mayfield and Black (2020) note that transformers such as BERT and DistilBERT performed similarly compared to the classical deep learning techniques, but seem to be able to generalize their results a bit better than the classical methods. They note that the lack of generalization across even transformers can be attributed to many models creating their own scores and values for certain features such as grammar and word diversity.

The most similar structure to this project comes from Yang et al. (2023), which focused on modeling on an ACT dataset to provide feedback for students. Yang's research group analyzed essays based on pre-labeled values that represent the different structures of an argumentative essay. This project's model is based on this technique of utilizing pre-labeled essay features, but the distinction comes from our attempt to solve the issue of generalization on essays that come from the same competition or standardized test.

3 Methodology

3.1 Task

The overall task of this work is to take an arbitrary essay and assign it a deserving, integer score that summarizes the overall quality of that essay. Our approach is to train a preliminary, first-stage model that will be able to extract useful sub-scores from essays and apply that model to previously unseen essays. The resulting sub-score predictions will then be used as features for a second-stage model that attempts to predict overall essay score. Each stage of the model will be trained on a different dataset. The first stage has six targets: the analytic measures of *cohesion*, *syntax*, *vocabulary*, *phraseology*, *grammar*, and *conventions*; these are the sub-scores referred to above. The second stage has a single target: overall essay score. Our expectation is that the first-stage will be able to accurately predict sub-scores for essays in the second dataset, and those predicted sub-scores will serve as useful features in the training of our second-stage.

3.2 Data

Our first-stage model is trained on a dataset provided by a 2022 English Language Learner's (ELL) competition (Franklin et al., 2022) in an attempt to alleviate the burnout that past teachers were suffering from hand grading essays. ELL's data contains around 4,000 essays written by 8-12th grade students. Each essay does not have a combined final score, but is graded based on the six categories mentioned above: *cohesion*, *syntax*, *etc*. These scores range from 1 (lowest) to 5 (highest) by increments of 0.5; so the nine valid scores are in the set {1, 1.5, . . . , 4.5, 5}.

A second dataset is fed into the first-stage model and is used for feature predictions. This second dataset comes from the notorious Kaggle competition (Hamner et al., 2011) conducted by the Hewlett Foundation in 2011 for 7th through 10th grade students. These essays are divided into 8 different "essay sets"; meaning that there are 8 unique essay prompts, each essay responds to one of the eight prompts. Each essay set has its own range of scores (the smallest range sees scores in 0-3 and the largest sees scores in 10-60). Both datasets are also split into 60% training, 20% validation, and 20% test sets. For the purposes of training time, this work only constructs models on a subset of the original 20000 total essays.

3.3 Preprocessing

Both essay datasets are separated word for word and converted into tensors. Punctuation and capitalization are both preserved because certain features such as grammar are conceptually related to these two factors. Some words in the Hewlett dataset have capital words with an '@' in front of them. These represent proper nouns that the students use in their essay and are removed from the inputted data because there is no clear indication of whether the student properly uses these terms or spells them correctly.

The ELL scores are transformed to range from 0 to 8 as integers instead of 1 to 5 to have a smoother transition of identifying whole number classes. Exploratory data analysis illustrates that middle classes like 3 and 4 are much more common than edge classes like 7 and 8. Even when we lump edge classes together, there is still a significant class imbalance. This will be addressed in the architecture.

The Hewlett Foundation dataset contains a wide range of scores for the different essay sets. We normalize all the score ranges so that each score is between 0 and 100 for training and then transform them back to their original score range for model evaluation.

3.4 Architecture

Our first-stage model operates by applying a base BERT model to an essay, then sending the outputs of the BERT model through a feed-forward neural network (Figure 1). Specifically, we extract the CLS token from the BERT model outputs, thus yielding a 768-dimensional feature vector to enter the feed-forward network. Exploratory data analysis indicates that in both datasets, over 80% of

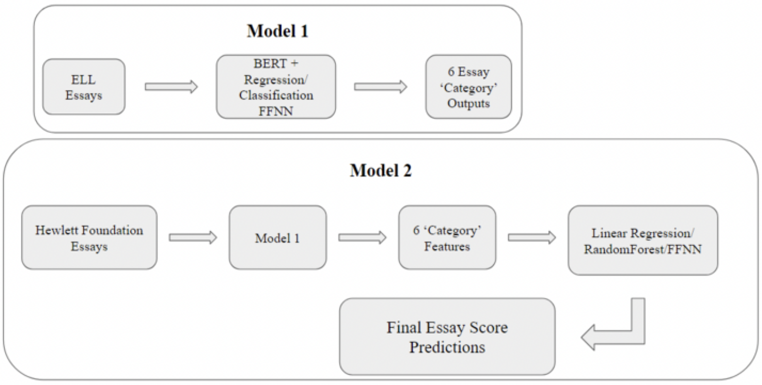
Model 1
ELL Essays
BERT + Regression/ Classification FFNN
6 Essay 'Category' Outputs
Model 2
Hewlett Foundation Essays
Model 1
6 'Category' Features
Linear Regression/ RandomForest/FFNN
Final Essay Score Predictions

Figure 1: Two Stage Model Architecture

essays have a word count of 600 or lower. Therefore, the max sequence length fed into BERT is 600 to compensate for training time.

The feed-forward network always consists of a single hidden layer (of varying size) and a fixed-size dropout layer. A crucial architectural detail is the structure of the six heads (one for each category) of the feed-forward network, given that the six targets are (discrete) ordinal variables. Ultimately, we implement two versions of the first-stage model: one version which treats scoring as a regression problem, and another version which treats scoring as a multi-class classification problem. The regression model is trained using Mean Squared Error as the loss function, whereas the classification model is trained using Sparse Categorical Crossentropy as the loss function. For the classification model, class weights are introduced to penalize incorrect predictions for minority class examples more heavily.

The outputs of the first-stage model are used as input features for the second-stage model. Because passing an essay through the first-stage model yields six sub-score predictions, the second-stage model accepts six input features.

Model 2 consists of a baseline and three other different architectures. All models have a linear output from 0 to 100 for ease of training, and then are transformed back to their own respective essay score ranges for evaluation. The baseline consists of finding a linear correlation between the word count of essays and the final score. Exploratory data analysis displays that there is a positive correlation between word count and final scores. This model is implemented to make sure that students who write the most words are not always going to receive the highest scores.

The initial model also utilizes a linear framework, but with a distinct approach: it incorporates the six sub-score predictions as inputs. In contrast, the second model adopts a random forest regression paradigm, allowing for the adjustment of hyperparameters such as max depth and the number of estimators. The final model takes a unique direction by channeling the six features through a feed-forward neural network (FFNN) to produce an encompassing essay score. This FFNN is designed with varying layer counts and hidden sizes, a strategy that acknowledges the diverse complexities of the feature scores. There is also a large number of hyperparameters such as learning rate and layer sizes. Training for every single hyperparameter combination consumes too much time and resources. This issue can be resolved by implementing a random search to ensure that a decent amount of combinations are trained while within an adequate time frame.

3.5 Evaluation Metric

The dominant evaluation metric used in the AES literature is the Quadratic Weighted Kappa (QWK) score, a version of Cohen's Kappa. The QWK score is appropriate when the target being analyzed is an ordinal variable. We choose to evaluate our overall models with this metric because the essay scores for dataset 2 are ordinal. We depart from QWK only in evaluating different iterations of our first-stage regression models, where we opt to use Mean Squared Error (MSE) because we want to preserve the continuous nature of the models' predictions.

Cohen's Kappa can also be interpreted as inter-rater reliability, or the agreement between two graders. This metric ranges from -1 to 1, where 1 represents complete agreement between two raters, 0 represents no agreement, and -1 represents complete disagreement. Therefore, the more an AES model's predictions agree with those of a human grader, the higher the QWK score will be. The formula for QWK can be seen below.

$$QWK = 1 - \frac{\sum_{i,j} W_{i,j} O_{i,j}}{\sum_{i,j} W_{i,j} E_{i,j}}$$

In this formula, each of W, O, and E are $N \times N$ matrices, where N is the number of values the variable can take on. $O_{i,j}$ represents the observed frequency with which rater 1 assigns score i and rater 2 assigns score j for $i, j \in N$; $E_{i,j}$ represents the expected frequency with which these same assignments would happen, assuming the raters assign scores *independently of each other*; $W_{i,j}$ represents the penalty applied to these assignments. The formula for $W_{i,j}$ can be seen below.

$$W_{i,j} = \frac{(i-j)^2}{(N-1)^2}$$

Note that the diagonal of W is populated with zeros, indicating that no penalty is applied when the two raters assign the same score. The penalty applied increases quadratically as the ratings become more dissimilar.

In order to arrive at a final evaluation of model performance, we must first calculate the QWK score within each essay set (because N varies across sets) and then average these scores. Mayfield and Black (2020) state that an acceptable performance will have a QWK within 0.5 and 0.8. Anything lower than this will not be considered "fit" to be able to assess human essays. In this case, an acceptable score of 0.5 or higher will be an acceptable model for predicting the second dataset.

4 Results

4.1 Model 1

When comparing itself to the baseline of 4, Table 1 indicates that the BERT Regression manages to almost cut the baseline average MSE in half at 0.947. The BERT Classification model manages to outperform its own baseline with an average QWK between the six scores at 0.583. The models with the specific hyperparameters that yielded these evaluation scores were used to run inference on the Hewlett dataset essays, thereby generating features for the second-stage model.

Model	Test Set Score
Baseline 4	MSE = 1.771
BERT Regression	MSE = 0.947
BERT Classification	QWK = 0.583

Table 1: Model 1 results. *This is the average QWK/MSE's across the 6 different essay aspects*

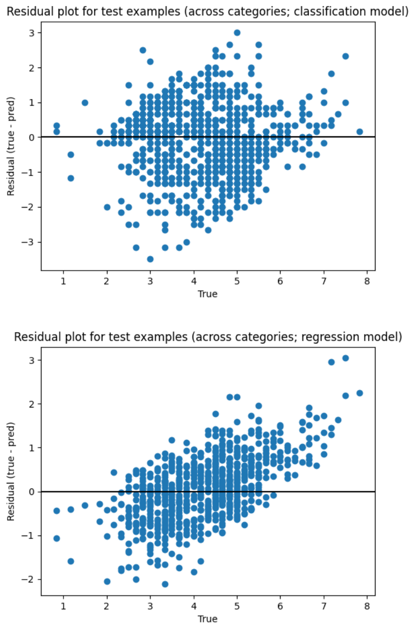

Figure 2: Residual Plots

In order to concisely grasp the errors of our first-stage models, we elect to display the residual plot of averages rather than a separate residual plot for each of the six categories. The plots in Figure 2 demonstrate that the regression model learns to be conservative in its predictions whereas the classification model performs fairly well with extreme values.

4.2 Model 2

As expected from the baseline, the average QWK across the eight essay sets with word count performs the worst at 0.154. Unfortunately, none of the models in Table 2 are able to reach the 0.5 average threshold for an acceptable model. The highest being the FFNN that takes in the predictions from the BERT Regression Model 1 at an average test QWK of 0.441. It is important to note that no matter the architecture for Model 2, the Model 1's Regression consistently surpasses its classification counterpart in terms of the average QWK across the different essay sets.

Model	Test Set QWK
Linear Word Count	0.154
BERT Classification + Linear	0.285
BERT Regression + Linear	0.396
BERT Classification + RF	0.355
BERT Regression + RF	0.377
BERT Classification + FFNN	0.340
BERT Regression + FFNN	**0.441**

Table 2: Model 2 results. *BERT Classification/Regression + Model means that either a BERT Classification or Regression is used to model the 6 essay features (grammar, cohesion, phraseology, vocabulary, syntax, conventions) and those 6 features were fed into the added Model. This table represents the average test QWK across all 8 essays in the Hewlett Foundation Dataset*

5 Discussion

Assigning a deserving score to an essay is no simple task. The complex structures and literary elements that generally give rise to quality writing can be hard to identify, even for human graders. For this reason, we decide to simplify the essay scoring problem by breaking it down into the two-stage process described above. We reason that predicting a grammar score, for example, should be easier than predicting an overall essay score directly; and that furthermore, this is likely similar to the process that a human grader follows anyway.

5.1 Experiments

Many potential paths remain even after deciding on the high level model architecture. Some decisions we make based on theory, and others we decide to experiment with to observe how different model choices influence the results. For example, the only encoder we use in model 1 is the BERT-base uncased model; time constraints dissuaded us from attempting other possible encoders at this stage in our pipeline. Such decisions that we put to experimentation include the choice of loss function for the first-stage model, whether or not to retrain any of the BERT layers, and what type of model to use in the second stage of our pipeline.

We settle on the BERT-base uncased model as our encoder because it is standard in the homework assignments and used in the AES field (Mayfield and Black, 2020). Given more time, we would like to explore how different choices of encoder for this initial step of model 1 would ultimately influence our results. Another important modeling choice that we make according to common convention is which of the BERT model outputs to use for subsequent layers of model 1. We use the CLS token to represent an essay because it is a good approximation of the semantic content of that essay. However, other possible choices at this juncture are to use the pooler token, or the context-dependent word embeddings directly.

As mentioned earlier, the targets used for training model 1 are ordinal variables, leading us to wonder which form of loss function is most appropriate for learning the relationships between the features extracted from BERT and the targets. We decide to implement two main versions of model 1 and then evaluate which leads to better overall model performance at the end of the whole pipeline. The first version treats the modeling task as a regression problem, using Mean Squared Error as the loss function and returning continuous-valued predictions as outputs. The second treated the modeling task as a multi-class classification task, using Sparse Categorical Crossentropy as the loss function and returning discrete-valued predictions as outputs. As the information type of the outputs are different, these two versions of model 1 beg for different evaluation metrics (MSE for the former and QWK for the latter) and therefore cannot be directly compared against essays in dataset 1. However, each version yields its own downstream performance (model 2 performance) and this performance can be directly compared. As noted in the results section, models which use regression in stage 1 perform superior to those which use classification in stage 1. This is likely because treating model 1 as a multi-class classification task essentially throws away information inherent in the ordinal nature of the target. Treating the task as a regression problem, on the other hand, allows the loss function to discriminate between "slightly wrong predictions" and "very wrong predictions".

We also test whether or not retraining any of the BERT layers improves model 1 performance. This project imparts on us the valuable lesson of considering resource consumption in making modeling decisions. Through a handy engineering trick, we found that - assuming we are not retraining any BERT layers - we can extract the CLS tokens from BERT in advance, and then use those as static inputs to the feed-forward networks of model 1 that follows. This trick enables us to bring the training time for a single epoch down to about 1 second.

In contrast, when we choose to retrain the outer 3 layers of the BERT model, training a single epoch takes a little under 4 minutes (about 200 times longer). Therefore, we are limited in terms of the number of epochs we can train for in the case of fine-tuning BERT. This limitation is what we believe leads to the severe under-performance of the fine-tuned versions of model 1 versus those which freeze all BERT layers. We choose not to feed any of the fine-tuned versions of model 1 into model 2 to complete the pipeline because we are confident that the overall performance would be inferior.

Lastly, we were unsure of the best way to use the features extracted from model 1 to arrive at a final overall essay evaluation, so we tried three variations: simple linear combination, a feed-forward neural network and a Random Forest model. We will comment on the latter two here. One of the main reasons the FFNN outperforms the Random Forest architecture is most likely due to the sample size. FFNN and Random Forest are both exceptional at capturing non-linear patterns, but the larger size of the second dataset could contribute to the Random Forest's lower QWK score. Another reason for the FFNN having a higher average QWK is likely due to it having a stronger representation of features. FFNNs thrive on learning from the data and use their hidden layers to capture more complex relations and abstract features in the data. Random Forests are limited to what is inputted into the model and may be limited in terms of training.

5.2 Error Analysis

We analyze the errors only for our stage 1 model because it represents the bulk of the natural language processing within this project. The residual plots seem to indicate that the classification model performs better than the regression model. This was surprising to us given the superior downstream performance of regression based stage 1 models. The conservative behavior of the regression model can probably be explained by the heavy imbalance of the target. We correct for this imbalance in the case of the classification model by using class weights to give stronger preference to minority predictions, but we do not take any corrective action in the case of the regression model. This behavior undermines the practical value of our model: we are consistently underestimating the quality of great essays and overestimating the quality of poor essays. We only became aware of this behavior very late in the course of our analysis, but given more time, we would have explored the use of a weighted MSE which would encourage prediction at the edges of the range of the target.

5.3 Limitations

We say that our model is prompt independent. By this, we mean that it can be applied to an arbitrary essay without accounting for the prompt or rubric that a student may have considered when writing the essay. This is good in the sense that it can work out of the box, without the need to be trained anew whenever it is used on a new essay set. It is bad in the sense that a student could hypothetically write about a topic completely unrelated to the prompt they are given and our model might give them a good grade. In future iterations of this work, we would like to explore the possibility of adding a mechanism to our stage 2 model that evaluates the degree to which an essay addresses its corresponding prompt, thus earning a "relevance" score in addition to the scores generated by model 1.

A potential weakness of our approach is that the model learned in stage 1 might not generalize to essays outside of dataset 1. From what we can tell, the essays of dataset 1 were all written by students learning English as a second language. Specifically, the six scores assigned to each essay in dataset 1 are meant to assess the "language proficiency" of the student that wrote it. For this reason, it seems plausible that the human graders evaluating these essays may have been focusing on different elements than the human graders evaluating the essays of dataset 2. In general, this begs the question of how many different valid ways there are for a human to grade an essay. This insight teaches us to take caution in making apples to apples comparisons of targets across datasets.

The final limitation we will discuss is information loss between stages of our model. The features used by the second stage of the model are synthetically generated data; as such, they contain a certain degree of noise. The stage-two model is therefore not receiving an essay directly, but rather a sort of first-order approximation of that essay that may or may not be accurate. Of course, the better our stage-one model, the more accurate these approximations are. We now suspect that the performance of our first-stage model was simply not good enough to generate useful features for our stage-two model.

6 Conclusion

Our conducted research aims to explore a new method for Automated Essay Scoring (AES) by using a two-stage model. The highest performing architecture is achieved utilizing a BERT Regression model to predict six essay feature scores as the first stage model. Inputting the predicted feature scores into a Feed Forward Neural network yields a test average QWK of .441. Nevertheless, it is important to note that even the best-performing two step architecture falls short of the 'acceptable' performance range of QWK between 0.5 and 0.8, indicating room for improvement.

Our research underscores the need for more datasets with essays that have been pre-labeled for different essay aspects. The aspects should not be limited to the six features that we have analyzed, but can include different essay features such as analyzing structure of an essay itself. Future models should include prelabeled features such as strength of claims, introductions, and evidence to further enhance the grading accuracy. Future studies can also expand on our research with more diverse language models. The different modifications of transformers such as ALBERT, XLNET, and RoBERTa have the potential to provide insights into how to create robust AES models.

Despite not reaching the desired performance, our study sheds some light on the importance of generalizing essay grading models. Our work contributes to the ongoing efforts in creating an efficient and unbiased AES model, and brings us closer to solutions for educational institutions requiring consistent essay evaluation.

References

Dimitrios Alikaniotis, Helen Yannakoudakis, and Marek Rei. 2016. Automatic text scoring using neural networks. pages 715–725.

Alex Franklin, Maggie, Meg Benner, Natalie Rambis, Perpetual Baffour, Ryan Holbrook, Scott Crossley, and ulrichboser. 2022. Feedback prize - english language learning.

Ben Hamner, Jaison Morgan, Lynn Vandev, Mark Shermis, and Tom Vander Ark. 2011. The hewlett foundation: Automated essay scoring.

Elijah Mayfield and Alan W Black. 2020. Should you fine-tune bert for automated essay scoring? pages 151–162.

Bingqing Yang, Sanghoon Nam, and Yutong Huang. 2023. "why my essay received a 4?": A natural language processing based argumentative essay structure analysis. 13916.

Katarina Zupanc and Zoran Bosnic. 2018. Increasing accuracy of automated essay grading by grouping similar graders. pages 1–6.