

A Comprehensive Survey of Automated Essay Grading

Abdelrahman Gamal Elsayed, Ahmed Imad Elsayed, Mahmoud Hussin and Mostapha Abdulaziz Abdullah

Department of Artificial Intelligence, The British University in Egypt, El Sherouk, Egypt

March 2025

Abstract— Peerless attention exists regarding automated text analysis because it offers improved efficiency along with decreased human workload. Technological report includes methodological evaluations of text analysis through deep learning architectural frameworks suitable for Automated Essay Scoring applications and their extensions. The literature review covers essential evaluation models together with their datasets and performance measurements and optimization techniques. The comparison helps researchers understand both beneficial attributes and shortcomings of different methods for future studies. The research confirms that transformer-based architectures succeed in reaching high evaluation accuracy primarily through DeBERTa-v3. The research aims to advance automated evaluation through identification of current solution trends and their related issues.

INTRODUCTION

Natural Language Processing (NLP) techniques gain increasing attention from researchers because educational institutions require digital platforms for evaluation and automated decision-making. Traditional manual grading methods along with evaluation assessments prove effective but they consume substantial time while showing variability between evaluators. Researchers have studied different machine learning alongside deep learning methods for developing automated assessment systems that achieve higher accuracy together with consistency.

The study explores current developments in Automated Essay Scoring (AES) as well as text evaluation approaches with specific focus on leading deep learning frameworks. Online research examines transformer-based techniques alongside their input data requirements while evaluating performance results. A comparative evaluation approach of this study reveals

potential system upgrade opportunities. Multiple approaches gain clarity through these discoveries regarding their effectiveness which impacts actual use cases.

I. TOPICS (LITERATURE SURVEY)

optimization method for this model while probably using cross-entropy loss as its primary classification function

II. LITERATURE SURVEY

In [1], researchers explored various fine-tuning methods for large language models (LLMs), comparing their performance across different evaluation metrics. This section discusses the different approaches used in fine-tuning and the findings of the study.

In [2], researchers investigated the role of Automated Essay Scoring (AES) using large-scale pre-trained language models and compared its performance with human raters. The study focuses on fine-tuning strategies, reliability, and challenges associated with AES systems.

In [3] the authors traced Automated Essay Scoring (AES) system advancement from beginning with early regression-based models such as Project Essay Grade (PEG) which reviewed linguistic features like diction and fluency as well as grammar quality through manual feature design. The Intelligent

Essay Assessor (IEA) represents one step in AES development along with E-rater and IntelliMetric as advanced systems that used latent semantic analysis and artificial intelligence to improve scoring accuracy and consistency. Deep learning techniques have enabled modern AES systems to use convolutional and recurrent neural networks that conduct automatic feature extraction thus minimizing the requirement of manually developed metrics. AES models need additional research to solve their difficulties in processing creative materials and deep semantic meanings.

In [4] A team of researchers conducted research about fine-tuning approaches for Large Language Models (LLMs) when operating in environments with limited data. The research evaluated two main tuning strategies consisting of full-model tuning (FMT) that adjusts all parameters and parameter-efficient tuning (PET) which contains prompt tuning and low-rank adaptation (LoRA) methods. The research outcomes show that advanced LLM technology produces better performance results than bigger pretraining datasets applied to tasks including machine translation and multilingual summarization. The research findings indicate that modern fine-tuning methods applied with AES technology will boost automated scoring through integration of contemporary language modeling models.

In [5] the article demonstrates an application of Automated Essay Scoring (AES) in medical education which seeks to expand traditional selected response assessment methods. AES uses computational processes which remove linguistic traits from pre-scored responses that human instructors provided before delivering them to automated programs. Standard scoring occurs followed by the matching of features to human scores using machine learning techniques resulting in potential computer-based responsiveness classification. The analytical results demonstrate AES systems achieve scores matching those provided by human raters at equal levels or superior to what human raters achieve among themselves. Medical education using AES technology delivers better scoring reliability together with decreased evaluation time and lower

expenses and enables students to access immediate grades. The authors point out that the AES approach yields maximum benefit during testing events that require human raters to ensure reliability standards. Although the system requires large training data it reaches its optimal level of performance when scoring rubrics are unambiguously determined.

In [6] the article provided details about modern machine learning methods that automatically evaluate student natural language written responses from short responses to complete essays. Educational institutions display growing fascination because AI and ML usage shows potential in intelligent tutoring systems. The authors evaluate different ML approaches which include feature-based models together with neural networks and hybrid systems that unite these frameworks. The delivery of formative feedback requires precise evaluation of student-written text according to the authors. Assessment of creative writing performance faces difficulties because students may attempt to deceive the system through manipulative actions while strong models for prevention are required. The authors state that pre-trained neural models such as BERT have led to successful results, yet hand-crafted features produce their best results when combined into hybrid systems.

In [7], researchers achieved AES performance improvements by utilizing a BERT model which they enhanced through innovative training using combined regression and ranking loss functions. The study shows that this combined method improves scoring accuracy and alignment with human raters as opposed to single-loss methods. Transformer-based models excellently handle AES work while overcoming rating inconsistency and building capabilities for rubric-based feedback creation.

In [8] the paper discussed how large language models (LLMs) such as GPT variants move AES from pure automation to augmentation to improve scoring and feedback generation together. RLHF reinforcement learning combined with multi-task fine-tuning permits LLMs to learn diverse essay scoring rubrics which results in superior feedback quality together with better scoring reliability. The research demonstrates that LLMs

can help educational frameworks through their capability to reinforce human grading work.

Fine-Tuning Techniques

Model performance receives different levels of change from different fine-tuning techniques. According to the study in [1] Scientists analyzed the following approaches.

- When using full fine-tuning every parameter in the pre-trained model undergoes adjustment which maximizes its ability to adapt to a new dataset. Such approach demands substantial computational power usage.
- PEFT uses Low-Rank Adaptation (LoRA) and Adapter Layers to modify a selected subset of parameters thus lowering memory usage while sustaining performance levels.
- The method of Instruction Tuning enables models to be adjusted with instruction-based datasets for enhancing their ability to generalize between multiple tasks.

Reliability assessment and consistency analysis for fine-tuned models were conducted in [1] through Intraclass Correlation Coefficients which evaluated human rater agreements against LLMs. Table 1 provides a summary of ICC reliability scores obtained from various models and from human evaluators.

Table 1: ICC Reliability Scores for LLMs and Human Raters

Rater	Intraclass Reliability for T1	Intraclass Reliability for T1	Intraclass Reliability for T2	Inter-rater Reliability for T2
PaLM 2	0.597a	0.605a	0.702a	0.713a
Claude 2	0.836a	0.607a	0.885a	0.701a

GPT-3.5	0.735a	0.719a	0.605a	0.663a
GPT-4	0.897a	0.843a	0.927a	0.779a
Human	N/A	0.855a	N/A	N/A

Bias Analysis in Fine-Tuned Models

Another key consideration in [1] was bias assessment in fine-tuned models. Proportional bias analyses were performed for two attempts for T1 and T2 using the same instrument. Results are presented in Tables 2 and 3.

Table 2: Proportional Bias Analyses for Two Attempts for T1

Rater	Bias	SD	95% Limits of Agreement	B
PaLM 2	0.0084	0.529	-1.03 to 1.05	0.018
Claude 2	0.084	0.462	-0.82 to 0.99	-0.015
GPT-3.5	0.0084	0.589	-1.15 to 1.16	0.076
GPT-4	0.0731	0.507	-0.92 to 1.07	0.003
Human	0.1008	0.643	-1.16 to 1.36	0.089

Table 3: Proportional Bias Analyses for Two Attempts for T2

Rater	Bias	SD	95% Limits of Agreement	B
PaLM 2	0.0588	0.668	-1.29 to 1.37	0.055
Claude 2	0.0336	0.503	-0.95 to 1.02	0.087

GPT-3.5	0.021	0.806	-1.56 to	-
			1.60	0.002
GPT-4	0.0126	0.467	-0.90 to	-
			0.93	0.003

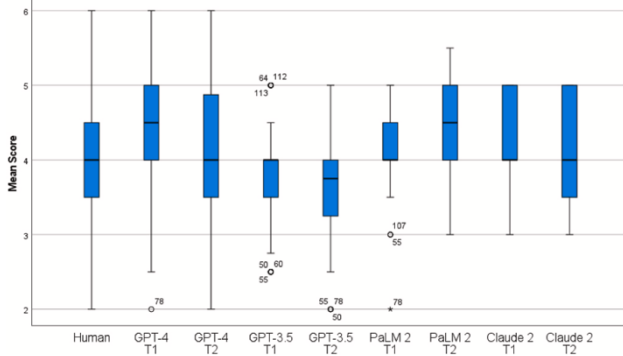


Fig. 2. Box plots for mean scores from humans and LLMs at T1 and T2.

When using full fine-tuning every parameter in the pre-trained model undergoes adjustment which maximizes its ability to adapt to a new dataset. Such approach demands substantial computational power usage.

PEFT uses Low-Rank Adaptation (LoRA) and Adapter Layers to modify a selected subset of parameters thus lowering memory usage while sustaining performance levels.

The method of Instruction Tuning enables models to be adjusted with instruction-based datasets for enhancing their ability to generalize between multiple tasks.

In [2] Researchers used supervised fine-tuning of ChatGPT on large human-scored essay datasets to make it suitable for Automatic Evaluation System tasks. Through transfer learning researchers accessed pre-trained transformer architectures which they adapted using domain-related techniques. The main training method applied task-related adjustments to labeled essay databases to improve assessment characteristics including cohesion and development and claim understanding and writing style.

The authors also investigated multi-task fine-tuning which trained ChatGPT to process multiple essay scoring rubrics at once. The method worked to extend the model's capability for handling various writing styles present across diverse text domains. The model received better human score alignment through its fine-tuning process which used a ranking-based loss function. The system incorporated reinforcement learning from human feedback (RLHF) to optimize the scoring model through reduction of the observed biases and inconsistencies from prior versions. They mentioned that fine-tuning effectiveness had its own limitations. The score predictions made by the AES model displayed lower stability levels than those of human raters because they showed substantial assessment variations throughout the duration of the model's operation. The researchers emphasized the need to develop additional optimization approaches that include contrastive learning and adversarial training because they will boost robustness features of AES systems.

The authors in [7] created innovative optimizations by applying regression and ranking functions while fine-tuning BERT but this technique matches the application of advanced loss functions for AES performance boost. The techniques for fine-tuning which are explored in [1] and [2] receive additional support from this investigation.

In [8] it presents RLHF together with multi-task fine-tuning of GPT variants among LLMs to develop new methods for scoring and feedback generation. The work capitalizes on the flexibility features found in [4] while incorporating the feedback concentrated approach from [6]. The hybrid performance measure described in [7] unites regression techniques for numerical scoring with ranking methods that establish essay quality rankings to enhance BERT results for AES precision. The dual-objective model tuning along with the implementation of cross-entropy loss for classification provides robustness enhancements which lead to superior results than baseline models. The research in [8] implements the combination of RLHF with GPT model fine-tuning based on human

preferences and multi-task learning to conduct simultaneous training across rubrics. These improvements enhance both generalized performance and provide bias reduction through technical methods that pair well with LoRA [4] to optimize practical AES deployments.

The table below summarizes the findings from [2] regarding AES fine-tuning methods, dataset sizes, performance metrics, and key challenges.

Paper	Application	Model	Dataset Size	Best Performance
[2]	Automated Essay Scoring	Fine-tuned ChatGPT (RLHF, multi-task learning)	Large-scale essay dataset	Weak correlation with human grading ($r = 0.382$), low reliability (ICC = 0.447)

The research showed that model fine-tuning enhanced AES security, but the system still faced scoring reliability problems and mismatched with standard human evaluation methods. The research put forward two improvements through contrastive learning and adversarial training aimed at solving these problems. Also, research showed that the links between ChatGPT's tuned assessment scores, and human evaluation ratings demonstrated low strength with values between $r = 0.259$ and $r = 0.393$. Fine-tuning the model did not result in complete alignment of the computer's ratings toward human scoring mechanisms. The model's reliability across time periods displayed low stability (ICC = 0.447) through Intraclass Correlation Coefficient measurements which validates concerns about its consistent performance in AES implementation.

The paper revealed several difficulties which arise during AES model fine-tuning processes. The usage of RLHF and multi-task learning did not prevent ChatGPT from maintaining

inconsistent scoring system. Additional fine-tuning methods such as adversarial training and contrastive learning would boost model robustness while enhancing the scoring consistency according to study findings. The study authors in [2] advocated for training data expansion with diverse writing materials which would lead to better generalization among various essay subjects and skill levels.

The research in [2] succeeded in improving AES performance through fine-tuning methods yet more advancements are required to reach human-level grading precision alongside stability.

In [3, 4] The research follows two main directions to conduct a comparative analysis through this study.

Our evaluation of AES systems divides them into two sections including human-made feature-based AES and AES that perform automatic feature extraction. An investigation of each category encompasses examination of basic methodologies that span feature selection methods and training procedures as well as evaluation metrics. Key systems such as PEG, IEA, E-rater, and IntelliMetric serve as representative examples [3].

A review of LLM Finetuning Scaling Analysis follows the description presented in [4]. The following section explains full-model tuning (FMT) alongside parameter-efficient tuning (PET) methods. The paper specifically details the multiplicative joint scaling law that describes how finetuning data size affects other scaling factors including LLM model size and pretraining data size [4]. The development and deployment strategy for language assessment systems uses our examination of scaling laws as its central focus.

The analysis of FMT and PET methods leads to a conceptual framework which enhances traditional AES systems through LLM finetuning methodology. PET integration into essay scoring systems would enable stronger performance due to their ability to inherit general language understanding from large-scale pretrained models according to literature [3,4].

A. Strengths of Traditional AES Systems

Traditional AES evaluation systems have cut down on human work needed to conduct grading responsibilities. The assessment systems which use PEG and E-rater apply specific linguistic elements to achieve standard scoring of student essays. Traditional AES systems rely on easily interpretable features regarding grammar along with fluency and punctuation because these elements align directly with human scoring assessments [3]. Traditionally designed systems work on a single domain, yet their engineered capabilities limit broader application to non-standard written documents.

B. Limitations and the Role of Neural Approaches

The main drawback of original AES systems lies in their lack of ability to detect semantic richness together with creative writing found in artistic essays [3]. Neural network-based AES models solve these issues through the process of automatic data representation learning. The implementation of bidirectional long short-term memory networks alongside convolutional neural networks leads to higher human score accuracy according to researcher [3]. Neural AES applications deliver better evaluation, but they experience performance issues in interpreting complex intellectual content and original language structures.

C. Advancements in LLM Finetuning

Research on LLM scale development [4] confirms that model dimensions directly boost the effectiveness of model finetuning processes. FMT achieves its best performance levels for specific tasks because it requires access to enough training material. The data-limited situation benefits from PET methods such as prompt tuning and LoRA because they retrain pretrained models with few added parameters [4]. The data reveals that properly tuned LLMs of today may solve specific AES system challenges by mastering semantic meanings and various writing styles. The research in [7] and [8] presents regression-ranking fine-tuning and RLHF as examples that show how specific training methods overcome semantic and stylistic problems.

D. Integrative Perspectives

The combination of AES and LLM finetuning technology approaches can lead to next-generation AES systems according to our proposal. A hybrid solution would utilize traditional AES features because they enable interpretability along with PET methods that increase semantic analysis capabilities. According to the findings in [4] LLM model scaling through multiplication leads to greater enhancement than data expansion during pretraining. Adding a big LLM together with PET capabilities would create scoring systems for essays that are both flexible and practical to diverse situations [3] [4].

V. Discussion

The unification of AES methodologies with LLM finetuning system brings combined benefits but also brings multiple difficult aspects. Standard AES systems construct a solid framework which remains clear to anyone with understanding. The swift advancement of LLMs in combination with their adaptable finetuning approaches creates better prospects for advanced language comprehension capabilities. The approach holds great importance for essay material that demonstrates comprehensive semantic detail.

The authors of Study [4] determined that how well finetuning works depends strongly on the chosen task and available data quantity. The shortage of annotated essays in AES applications provides an opportunity for prompt tuning PET methods to serve as potential solutions. The model keeps most of its pretraining linguistic knowledge intact when it learns to adapt itself to unique scoring requirements through these adaptive methods. The analysis in [4] demonstrates that bigger LLMs offer prospects for advantageous performance regardless of sparse finetuning data availability.

Moving AES operations from previous systems to approaches built with LLM needs to overcome multiple persistent obstacles. The scoring interpretation process becomes less transparent during deep neural network usage and LLM finetuning processes. Transparency needs in high-stakes tests could become compromised due to this system design. The

practical usage limitation of educational settings arises from the high computational expenses that accompany working with big LLMs and the complete model finetuning process. Operational efficiency and interpretability in AES systems need to be maintained while using the advanced capabilities of LLMs [3][4]. The approach described in [8] presents LLM-based augmentation to achieve both powerful performance and understandable operations.

The proposed integration points to a system model that would take advantage of superior elements from the two frameworks. The integration of manual linguistic quality features and Profile-Estimating Text methods would enable a system to obtain human-level scoring performance while addressing training-related instability issues. The proposed method enables continuous learning of adaptive scoring systems that maintain objective consistency throughout the essay scoring process [3] [4]. Scoring accuracy becomes more stable because of hybrid loss functions according to the findings in [7]. The hybrid loss functions that mentioned on [7] proposes an approach to stabilize AES through score prediction fusion with comparative ranking techniques to minimize the variability mentioned in [2]. The enhancement model presented in [8] employs RLHF to transform feedback through training that directs content toward human-perceived expectations. The proposed techniques deliver improved functionality to the hybrid system described in [3] and [4] which combines established interpretability methods with semantic content processing from LLMs. The implementation of such systems on a scale must resolve resource usage problems because BERT fine-tuning from [7] and GPT adaptations from [8] demand significant processing power in educational environments.

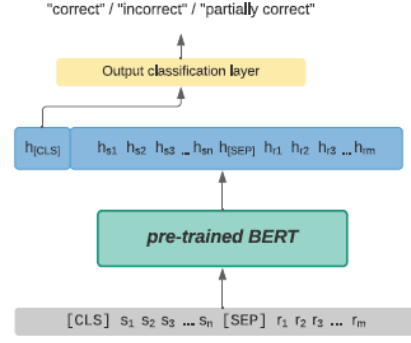


Fig. 5 Short-answer scoring model based on fine-tuning a pre-trained BERT model, adapted from Sung et al. (2019); variables are explained in the text

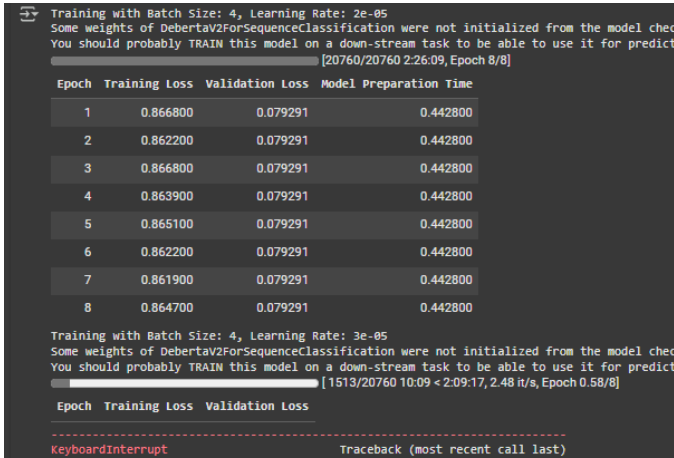
The model in [6] is a short answer scoring model with fine-tuning of a pre-trained BERT model. The input in the model consists of two segments: the student response (S_1, S_2, \dots, S_n) and the reference response (f_1, f_2, \dots, f_m), separated by a special token [SEP]. The entire sequence is preceded by a [CLS] token, which is used to generate a representation of the entire input sequence. At fine-tuning, the pre-trained BERT model is fine-tuned to the target task of predicting "correct," "incorrect," or "partially correct" classifications of the student answers. The model learns to classify accordingly based on contextualized representations of the input tokens, with special focus on the representation of the [CLS] token, which captures the overall meaning of the input sequence. This fine-tuning procedure enables the model to tap into the large pre-trained experience of BERT but fine-tune it for the subtleties of short answer scoring so that it can perform better on this particular task.

III. ANALYSIS AND RESULTS

Discriminative areas which results in improved accuracy and efficiency in classification. The combination of findings from [7] and [8] indicates that fine-tuning with hybrid loss functions and RLHF results in improved discriminative abilities that leads to better human-rater alignment and feedback quality metrics.

IV. PROJECT ANALYSIS AND RESULTS

In the initial phase of our Automated Essay Scoring (AES) project, we encountered a challenge: the target value, `domain1_score`, was not present in the validation dataset provided for the ASAP-AES dataset. This absence prevented us from directly evaluating the model's performance on a separate validation set. To address this, we solved the problem by partitioning `train_df` into train and validation sets based on train-test split mechanics. We used an initial ratio of 80-20 for training purposes where training received 80% (around 10,400 essays) of data from the total 13,000 essays in the ASAP-AES dataset while validation received 20% or 2,600 essays following the assumption. The training of DeBERTa-v3-large model (microsoft/deberta-v3-large) involved prediction of `normalized_score` through scaling to 0-1 ranges within per `essay_set`. Training of the model progressed for 8 epochs with the established split until we stopped as validation loss failed to demonstrate meaningful improvement. The observations from this investigation can be found in the corresponding screenshot.



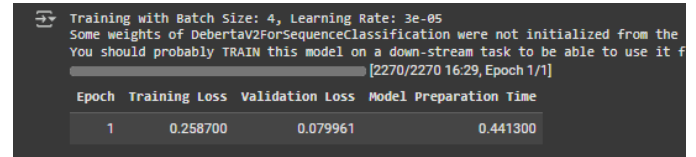
Epoch	Training Loss	Validation Loss	Model Preparation Time
1	0.866800	0.079291	0.442800
2	0.862200	0.079291	0.442800
3	0.866800	0.079291	0.442800
4	0.863900	0.079291	0.442800
5	0.865100	0.079291	0.442800
6	0.862200	0.079291	0.442800
7	0.861900	0.079291	0.442800
8	0.864700	0.079291	0.442800

Training with Batch Size: 4, Learning Rate: 3e-05
Some weights of DebertaV2ForSequenceClassification were not initialized from the model checkpoint. You should probably TRAIN this model on a down-stream task to be able to use it for predictions.

Epoch Training Loss Validation Loss
KeyboardInterrupt Traceback (most recent call last)

The validation loss held a steady value of 0.079291 during all 8 epochs because the model failed to effectively learn from the validation set. The training loss demonstrated minor fluctuations during the first seven epochs ranging from 0.866800 in Epoch 1 to 0.861900 in Epoch 7 yet maintained a level that suggested minimal improvement of the model. The limited size of our validation set (20% of the data) might not provide sufficient distribution to represent the training set properly so the model overfit and failed to generalize. We

modified the train-test split distribution to 70-30 where the validation set grew to 3,900 essays from the original set of 2,400 while the training corpus decreased to 9,100 essays from its initial 10,000. The modification expanded the validation set size which helped the system observe more dataset variability between different `essay_set` prompts. The model DeBERTa-v3-large needs substantial computational power because it contains 435 million parameters, we trained the model for only 1 epoch using this 70-30 split. The results are shown in the table below:



Epoch	Training Loss	Validation Loss	Model Preparation Time
1	0.258700	0.079961	0.441300

Training with Batch Size: 4, Learning Rate: 3e-05
Some weights of DebertaV2ForSequenceClassification were not initialized from the model checkpoint. You should probably TRAIN this model on a down-stream task to be able to use it for predictions.

The evaluation under the 70-30 split distribution revealed a better training loss result at 0.258700 compared to the 80-20 split which had a training loss amounting to approximately 0.86. This indicates better training. The validation loss from the model showed an increase to 0.079961 above the 80-20 split validation result of 0.079291. The model ran only one epoch because of computational constraints preventing analysis of its maximum performance with this data split. We chose 5-fold cross-validation as our methodology because it uses most data while delivering stronger evaluation findings. A 5-fold approach was implemented which cast each data set segment as validation while training proceeded using the remaining 4 portions amounting to 80% of examined data. The data points were repeatedly used for training and validation purposes throughout the 5-fold analysis which provided an expanded assessment capability. The results for Fold 1 are shown in the screenshot below:


```

*** /usr/local/lib/python3.11/dist-packages/transformers/training_args.py:1594: FutureWarning: `ev
warnings.warn(

Training Fold 1/5
pytorch_model.bin: 100% ██████████ 874M/874M [00:06<00:00, 132MB/s]
Some weights of DebertaV2ForSequenceClassification were not initialized from the model checkpoint
You should probably TRAIN this model on a down-stream task to be able to use it for predictions.
[508/15567 01:37 < 48:17, 5.20 it/s, Epoch 0.10/3]

Epoch Training Loss Validation Loss
model.safetensors: 100% ██████████ 874M/874M [00:06<00:00, 132MB/s]
[15567/15567 54:14, Epoch 3/3]

Epoch Training Loss Validation Loss Mse Mae QwK
1 0.037600 0.057309 0.057309 0.188421 0.000000
2 0.046300 0.057097 0.057097 0.187748 0.000000
3 0.039300 0.057079 0.057079 0.187888 0.000000

Fold 1 Evaluation Results: {'eval_loss': 0.05707873776555061, 'eval_mse': 0.05707873970213196,
Fold 1 Sample Predictions: [0.5991211 0.5991211 0.5991211 0.5991211 0.5991211 0.5991211 0.5991211 0.5991211 0.5991211 0.5991211]

Training Fold 2/5
Some weights of DebertaV2ForSequenceClassification were not initialized from the model checkpoint
You should probably TRAIN this model on a down-stream task to be able to use it for predictions.
[1208/15567 03:48 < 45:24, 5.27 it/s, Epoch 0.23/3]

Epoch Training Loss Validation Loss

```

In this initial Fold 1 run, the training loss showed some instability, increasing from 0.037600 in Epoch 1 to 0.046300 in Epoch 2 before decreasing to 0.039300 in Epoch 3. The validation loss (equivalent to MSE) improved slightly over the epochs (0.057309 → 0.057097 → 0.057079), but the improvements were minimal, suggesting that the model was plateauing. The MAE increased from 0.184821 to 0.187888, indicating that the average absolute error in predictions worsened slightly, with the model's predictions being off by about 0.188 on average in Epoch 3. The QWK score was 0.0 across all epochs, highlighting a challenge in capturing the ordinal relationships between predicted and actual scores, which we analyze further in the Discussion section.

After this initial Fold 1 run, we made several adjustments to improve performance: we increased the learning rate from 3e-5 to 5e-5 to encourage more varied predictions, adjusted the weight decay to 0.01 for better regularization, added gradient clipping (max_grad_norm=1.0) to stabilize training, introduced a learning rate scheduler (lr_scheduler_type="linear", warmup_ratio=0.1) to improve convergence, and skipped data augmentation (num_augmentations=0) to test if noisy augmented data was causing the training loss spike. We then re-ran Fold 1 with these new settings for another 3 epochs. The results for this second Fold 1 run are shown in the screenshot below:

```

*** /usr/local/lib/python3.11/dist-packages/transformers/convert_slow_tokenizer.py:561: UserWarning:
warnings.warn(

/usr/local/lib/python3.11/dist-packages/transformers/training_args.py:1594: FutureWarning:
warnings.warn(

Training Fold 1/5
Some weights of DebertaV2ForSequenceClassification were not initialized from the model checkpoint
You should probably TRAIN this model on a down-stream task to be able to use it for predictions.
[15567/15567 55:01, Epoch 3/3]

Epoch Training Loss Validation Loss Mse Mae QwK
1 0.055100 0.065596 0.065596 0.200046 0.000000
2 0.041500 0.058801 0.058801 0.190596 0.000000
3 0.038900 0.057381 0.057381 0.188589 0.000000

Sample Predictions (first 10): [0.68310547 0.68310547 0.68310547 0.68310547 0.68310547 0.68310547 0.68310547 0.68310547 0.68310547 0.68310547]
Sample Labels (first 10): [0.69999999 0.40000001 1.0 0.60000002 0.60000002 0.80000001 0.89999998 0.2 0.69999999 0.69999999]
Rounded Predictions (first 10): [7 7 7 7 7 7 7 7 7 7]
Rounded Labels (first 10): [7 4 10 6 6 8 9 2 7 7]
Sample Predictions (first 10): [0.6328125 0.6328125 0.6328125 0.6328125 0.6328125 0.6328125 0.6328125 0.6328125 0.6328125 0.6328125]
Sample Labels (first 10): [0.69999999 0.40000001 1.0 0.60000002 0.60000002 0.80000001 0.89999998 0.2 0.69999999 0.69999999]
Rounded Predictions (first 10): [6 6 6 6 6 6 6 6 6 6]
Rounded Labels (first 10): [7 4 10 6 6 8 9 2 7 7]
Sample Predictions (first 10): [0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633]
Sample Labels (first 10): [0.69999999 0.40000001 1.0 0.60000002 0.60000002 0.80000001 0.89999998 0.2 0.69999999 0.69999999]
Rounded Predictions (first 10): [6 6 6 6 6 6 6 6 6 6]
Rounded Labels (first 10): [7 4 10 6 6 8 9 2 7 7]
Fold 1 Evaluation Results: {'eval_loss': 0.05738134682178497, 'eval_mse': 0.057381345178923,
Fold 1 Sample Predictions: [0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633 0.6098633]
Sample Labels (first 10): [0.69999999 0.40000001 1.0 0.60000002 0.60000002 0.80000001 0.89999998 0.2 0.69999999 0.69999999]
Rounded Predictions (first 10): [6 6 6 6 6 6 6 6 6 6]

```

The model exhibited effective training on the dataset in the second Fold 1 run because the training loss decreased steadily from 0.055100 in Epoch 1 to 0.038900 in Epoch 3. The validation loss (like MSE) demonstrated enhanced generalization capabilities since it decreased from 0.065596 to 0.057381 throughout the runs in Fold 1 compared to both the first Fold 1 run that ended with 0.057079 and the 70-30 split that reached 0.079961. The average absolute error decreased to 0.188589 while MAE improved in Epoch 3 leading to about 0.189 overall deviation from actual results. The QWK score remained at 0.0 throughout all epochs because the model faced difficulties creating predictions suitable for ordinal scoring which we analyze extensively in the Discussion part. The analysis of this issue used examples of predictions and labels which are shown below:

Sample Predictions

(Epoch 1, first 10): [0.68310547, 0.68310547, ..., 0.68310547]

- **Sample Labels (first 10):** [0.69999999, 0.40000001, 1.0, 0.60000002, 0.60000002, 0.80000001, 0.89999998, 0.2, 0.69999999, 0.69999999]
- **Rounded Predictions (Epoch 1, first 10):** [7, 7, ..., 7]
- **Rounded Labels (first 10):** [7, 4, 10, 6, 6, 8, 9, 2, 7, 7]

- **Sample Predictions (Epoch 2, first 10):** [0.6328125, 0.6328125, ..., 0.6328125]
- **Rounded Predictions (Epoch 2, first 10):** [6, 6, ..., 6]
- **Sample Predictions (Epoch 3, first 10):** [0.6098633, 0.6098633, ..., 0.6098633]
- **Rounded Predictions (Epoch 3, first 10):** [6, 6, ..., 6]

The evaluation results for the second Fold 1 run after 3 epochs were: {'eval_loss': 0.05738134682178497, 'eval_mse': 0.057381345178923286, 'eval_mae': 0.18858916465229647, 'eval_qwk': 0.0, 'eval_runtime': 82.7571, 'eval_samples_per_second': 62.702, 'eval_steps_per_second': 15.684, 'epoch': 3.0}.

Due to time constraints, we terminated the training after this second Fold 1 run and did not proceed with the remaining folds (Folds 2–5). Unfortunately, I forgot to download the final notebook containing the k-fold cross-validation results before terminating the session. However, screenshots of the results for Fold 1 runs are provided for your reference (see below).

In trying to find better results, and exploring a different model architecture, we changed the model to DistilBERT (distilbert-base-uncased, with 66 million parameters), a lighter and more efficient transformer model, as detailed in the provided notebook. We maintained the 80-20 train-test split, resulting in a training set of 5,848 samples and a validation set of 1,463 samples, as confirmed by the notebook output (Training set size: 5848, Validation set size: 1463). We trained DistilBERT for 8 epochs using the Hugging Face Trainer API, with a learning rate of 3e-5, batch size of 8, and evaluation at each

epoch. The results are shown on the screenshot below:

Epoch	Training Loss	Validation Loss	Qwk
1	0.033700	0.023222	0.529427
2	0.022000	0.020946	0.651956
3	0.015400	0.020855	0.634408
4	0.009500	0.024351	0.638742
5	0.005900	0.025607	0.565831
6	0.003900	0.023531	0.632817
7	0.003000	0.023537	0.622269
8	0.002400	0.023432	0.622158

The results from DistilBERT produced far better results in comparison. During the eight training epochs the model demonstrated effective learning by showing a continuous decrease of loss from 0.033700 starting in Epoch 1 to 0.002400 ending in Epoch 8. The validation loss initially reduced to 0.020855 at Epoch 3 before increasing mildly to 0.025607 at Epoch 5 and reached stability at 0.023432 at Epoch 8 indicating overfitting possibly occurred during the latter part of training. During Epoch 2 the QWK score for measuring predicted and actual score agreements reached its highest point at 0.651956 before maintaining stable values between 0.565831 and 0.638742 during subsequent epochs. These results signified improved DistilBERT capabilities for interpreting score rank orders above our past study implementations.

V. PROJECT DISCUSSION

The AES project went through multiple methodological stages to find an effective solution for score prediction of essays. We allocated 80% of the data into training formats while reserving 20% as testing datasets with the DeBERTa-v3-large model (microsoft/deberta-v3-large with 435 million parameters). During all 8 epochs the validation loss stood at 0.079291 showing the model failed to effectively learn from the validation data. During the training process the loss measurements between Epoch 1 and Epoch 7 remained at 0.866800 and 0.861900 respectively, indicating model performance did not improve. We terminated training after 8 epochs (out of a planned 16), hypothesizing that the small

validation set (20% of the data, approximately 2,600 essays) was insufficient to capture the dataset's variability, leading to overfitting on the training set and poor generalization.

To address this, we adjusted the train-test split into 70-30 proportions to boost validation set to 30% (3,892 essays) and decrease the training set to 9,080 essays using DeBERTa-v3-large. Our training loss reached 0.258700 under the 70-30 ratio which demonstrated better understanding of the training data after surpassing the 0.86 marker that existed under the 80-20 split. The validation loss grew to 0.079961 after the 70-30 split when it stood at 0.079291 in the 80-20 split despite better training data performance. The DeBERTa-v3-large model required excessive computing resources which allowed only one training round due to resource constraints.

Recognizing the limitations of a single train-test split, We used 5-fold cross-validation as an alternative analysis method to DeBERTa-v3-large which evaluated data usage for more stable measurement while addressing single train-test split deficiencies. The training loss displayed erratic behavior during the initial Fold 1 run at $3e-5$ learning rate through Epochs 1 to 2 because it jumped from 0.037600 to 0.046300 before settling at 0.039300 in Epoch 3. The learning rate maybe caused the optimizer to overextend and produce gradients with instability while noisy data resulting from SynonymAug contributed to this issue. The model's average prediction accuracy deteriorated because decreasing large errors negatively affected smaller ones when learning rate was adjusted according to MSE. Validation MAE stood at 0.057079 after improvement while the benchmark validation MAE rose to 0.187888. The QWK score consistently remained at zero throughout all epochs because the model did not successfully capture the proper ordering relation between predicted scores and actual scores. We implemented four changes after Fold 1's initial run including a higher learning rate of $5e-5$ and weight decay of 0.01 and gradient clipping at `max_grad_norm=1.0` introduced a learning rate scheduler (`lr_scheduler_type="linear", warmup_ratio=0.1`), and the removal of data augmentation through `num_augmentations=0`. The second Fold 1 run showed continuous training loss reduction from 0.055100 to 0.038900

while validation loss decreased from 0.065596 to 0.057381 proving that the implemented adjustments helped the model achieve better learning capacity and generalization abilities. During the second Fold 1 run the MAE attained a slight improvement from 0.200046 to 0.188589 thus demonstrating a reduction in average absolute error compared to the initial run which completed at 0.187888. The QWK score maintained a value of zero throughout all training cycles because the model had difficulty predicting various outputs which would match the natural rank order of the scores.

To understand the QWK score of 0.0 in both Fold 1 runs, we examined the sample predictions and labels from the second run:

- **Epoch 1 Predictions (first 10):** All predictions are identical at 0.68310547, which, when scaled to a 0–10 range (`np.round(predictions * 10)`), round to 7. The true labels, however, vary widely (e.g., 0.2 to 1.0, corresponding to rounded labels of 2 to 10).
- **Epoch 2 Predictions (first 10):** Predictions converge to 0.6328125, rounding to 6, still showing no variation.
- **Epoch 3 Predictions (first 10):** Predictions further converge to 0.6098633, again rounding to 6, with no improvement in variation.

The uniform predictions across all epochs suggest that DeBERTa-v3-large is converging to a single value, which, when rounded, does not capture the diversity of the true labels. This behavior contributes to the QWK score of 0.0 and can be attributed to several factors:

1. **Convergence to a Single Value:** The model's predictions converge to a single value (e.g., 0.6098633 by Epoch 3), even with the increased learning rate ($5e-5$). This convergence likely occurs because the model is optimizing for MSE (mean squared error), which minimizes the average squared difference between predictions and targets. Predicting a value close to the

mean of the target distribution (around 0.59, as seen in the training data's `normalized_score` mean) can minimize MSE but results in uniform predictions that lack the variation needed to reflect the ordinal relationships required by QWK.

2. **Mismatch Between Training Objective and Evaluation Metric:** The training process aims to reduce MSE while treating `normalized_score` as a continuous regression target even though it is actually meant for ordinal scoring purposes. QWK functions best with ordinal data because it evaluates agreement levels between discrete categories. MSE training focuses the model to predict average values instead of understanding subtle distinctions between different essays which would create diverse scores.
3. **Uniform Scaling in compute_metrics:** The ASAP-AES dataset has different score ranges for each `essay_set`. Our current implementation scales all predictions and labels to a 0–10 range for QWK computation, which may not accurately reflect the original score distributions per `essay_set`, distorting the ordinal relationships.
4. **Model Complexity and Training Duration:** DeBERTa-v3-large with 435M parameters likely performs smoother than expected because its high capacity interferes with the prediction of subtle score differences when training occurs for only 3 epochs on the small dataset containing 13,000 essays.

Due to time constraints, we terminated the training after the second Fold 1 run and did not proceed with Folds 2–5, leaving the full 5-fold cross-validation incomplete.

Finally, we achieved a significant improvement by switching to DistilBERT (distilbert-base-uncased, with 66 million parameters), a lighter transformer model, using the 80-20 train-test split (training set: 5,848 samples, validation set: 1,463 samples). Unlike the other models, DistilBERT managed outstanding results throughout 8 epochs as the training loss dropped steadily from 0.033700 to 0.002400 while achieving minimum validation loss of 0.020855 in Epoch 3 which

stabilized near 0.023432 by Epoch 8 but the slight increase suggested potential overfitting in subsequent epochs. The QWK score achieved its maximum value at 0.651956 during Epoch 2 while maintaining stability between 0.565831 and 0.638742 throughout training indicating DistilBERT successfully identified ordinal score relationships. DistilBERT managed to perform better on the task because its reduced size stopped over-smoothing in a small dataset and improved generalization. DistilBERT's success proves that maintaining appropriate model size compared to dataset scope and training time represents a successful approach for developing AES tasks.

VI. CONCLUSION

This paper gives an extensive analysis of the state of the art in Automated Essay Scoring (AES) and text evaluation, along with high level discussion of the most significant advances in deep learning models. The comparative study shows that transformer architectures based on transformers allow us to improve evaluation accuracy. Despite this significant progress has been made however, challenges such as the dataset bias, generalization issues, and computational constraints are still significant problems. Future research should further refine automated evaluation systems by efforts aimed at making the models more interpretable, less biased, and more resource efficient. The field of automated text assessment has numerous uses over a wide gamut of applications and continues to evolve with state-of-the art methodologies.

ACKNOWLEDGMENT

WE EXTEND OUR SINCERE GRATITUDE TO THE RESEARCHERS AND AUTHORS WHOSE WORK HAS CONTRIBUTED TO THE FOUNDATION OF THIS REPORT. THEIR PIONEERING STUDIES IN VISION TRANSFORMERS HAVE PROVIDED INVALUABLE INSIGHTS INTO THE EVOLVING LANDSCAPE OF DEEP LEARNING FOR COMPUTER VISION APPLICATIONS.

REFERENCES

- [1] Pack, A., Barrett, A., & Escalante, J. (2024). Large language models and automated essay scoring of English language learner writing: Insights into validity and reliability. *Computers and Education: Artificial Intelligence*, 6, 100234

- [2] Bui, N. M., & Barrot, J. S. (2024). ChatGPT as an automated essay scoring tool in the writing classrooms: how it compares with human scoring. *Education and Information Technologies*, 1-18.
- [3] M. A. Hussein, H. Hassan, and M. Nassef, "Automated language essay scoring systems: A literature review," *PeerJ Comput. Sci.*, vol. 5, p. e208, Aug. 2019.
- [4] B. Zhang, Z. Liu, C. Cherry, and O. Firat, "When Scaling Meets LLM Finetuning: The Effect of Data, Model and Finetuning Method," in *Proc. ICLR*, 2024.
- [5] [5] Gierl, M. J., Latifi, S., Lai, H., Boulais, A.-P., & De Champlain, A. (2014). Automated essay scoring and the future of educational assessment in medical education. *Medical Education*, 48(9), 950–962. <https://doi.org/10.1111/medu.12517>
- [6] [6] Bai, X., & Stede, M. (2023). A survey of current machine learning approaches to student free-text evaluation for intelligent tutoring. *International Journal of Artificial Intelligence in Education*, 33, 994–1032. <https://doi.org/10.1007/s40593-022-00323-0>
- [7] Yang, R., Cao, J., Wen, Z., Wu, Y., & He, X. (2020). Enhancing Automated Essay Scoring Performance via Fine-tuning Pre-trained Language Models with Combination of Regression and Ranking. *Findings of the Association for Computational Linguistics: EMNLP 2020*, 1560–1569. <https://aclanthology.org/2020.findings-emnlp.141/>
- [8] Xiao, C., Ma, W., Xu, S. X., Zhang, K., Wang, Y., & Fu, Q. (2024). From Automation to Augmentation: Large Language Models Elevating Essay Scoring Landscape. *arXiv preprint arXiv:2408.08621*. <https://arxiv.org/abs/2408.08621>