IntelliGrade: Enhancing Descriptive Answer Assessment with LLMs

Amrit Gupta
Khoury College of Computer Sciences
Northeastern University
Boston, U.S.A.
gupta.amr@northeastern.edu

Aswath Senthil Kumar
Khoury College of Computer Sciences
Northeastern University
Boston, U.S.A
senthilkumar.as@northeastern.edu

Vishant Ketan Mehta
Khoury College of Computer Sciences
Northeastern University
Boston, U.S.A
mehta.visha@northeastern.edu

Akshant Alkesh Gandhi
Khoury College of Computer Sciences
Northeastern University
Boston, U.S.A
gandhi.aks@northeastern.edu

***Abstract*—Automated assessment of student answers presents a promising avenue for expediting grading processes and enhancing the overall learning experience. This project focuses on leveraging machine learning techniques, particularly Large Language Models (LLMs), to evaluate descriptive answers. By incorporating advanced natural language processing (NLP) methodologies with feature engineering and model fine-tuning, the system aims to provide nuanced scores reflective of the answers' quality and relevance to the posed questions. We strive to develop a robust and versatile system capable of delivering constructive feedback to educators and learners alike. Our objective is to streamline the grading workflow, enrich educational experiences, and set new standards in automated assessment practices.**

***Index Terms*—automated assessment, machine learning, large language models (LLMs), natural language processing (NLP), short answer evaluation, feature engineering, model fine-tuning**

I. Introduction

In the realm of educational assessment, the traditional manual grading process often poses challenges in terms of subjectivity and time consumption. The advent of machine learning and natural language processing technologies presents an opportunity to revolutionize this landscape by automating the evaluation of student answers. This project endeavors to capitalize on these advancements by developing an automated system for assessing descriptive answer. By harnessing the power of pre-trained Large Language Models (LLMs) and incorporating additional features extracted from the text data, we aim to provide holistic evaluations that go beyond mere grammatical correctness to assess coherence, relevance, and depth of understanding.

The core task revolves around designing a scoring mechanism that captures the essence of students' responses, aligning them with the underlying learning objectives. Through rigorous model training and fine-tuning on the manually curated dataset, our system endeavors to emulate the nuanced judgment of human graders, but at scale and with enhanced efficiency. Leveraging techniques such as feature engineering and iterative refinement, we seek to continuously enhance the model's comprehension and evaluation capabilities.

Furthermore, the project emphasizes not only the technical aspects of model development but also the practical integration of the automated assessment system into educational workflows. By providing a user-friendly interface and interpretability features for educators, we aim to foster trust in the automated grading process and facilitate meaningful feedback exchanges between educators and learners. Through these concerted efforts, we aspire to contribute to the advancement of educational assessment practices, ultimately enriching the teaching and learning ecosystem.

II. About the data

In our mission to revolutionize descriptive answer assessment, we performed an extensive exploration of diverse datasets relevant to question-answering contexts. We delved into the MS Marco dataset from Microsoft, the Natural Questions dataset from Google, the Pubmed QA dataset, and the MASH-QA dataset.

A. MS Marco

The MS Marco dataset [1] includes a large scale collection and with a primary goal of providing advanced deep learning techniques whilst performing Information Retrieval applications. It encompasses other sub datasets within such as the Question Answering (QnA) V1 and V2 dataset, which provides replies depending the inquiries by the user and the responses created by humans. Furthermore, it can be observed that datasets for passage ranking , key phrase extraction and conversational search situations are included. MS MARCO has accumulated over a million inquires and an extensive array of tasks, making it a significant resource for academics and practitioners in the domains of question answering, natural

language comprehension, information retrieval. Unfortunately, most of the answers given in the dataset were shorter and not descriptive enough which did not fit very well with the goals of our project.

B. Natural Questions

The Natural Questions [8] dataset is a benchmark dataset released by Google for the task of question answering. It consists of real user queries issued to the Google search engine along with corresponding Wikipedia passages that may contain the answer to the query. The dataset includes both long and short answers, along with annotations indicating the correctness and relevance of each answer. Initially, we intended to utilize Wikipedia passages as context for scoring answers. However, we found that the passages often contained excessively detailed and broad information that extended well beyond the scope of the questions. This surplus of information could potentially lead to inaccurate scoring, as the model might struggle to focus on the relevant details. Consequently, it was not integrated into our dataset for training and evaluation purposes.

C. PubMedQA

A particular dataset created for biological question answering is the PubMedQA [6] dataset. It includes questions based on actual user searches for biomedical literature and PubMed abstracts. The dataset includes information on diseases, treatments, and biological processes, among many other biomedical topics. Despite providing a context field, similar to the challenges we faced with the Natural Questions Dataset, the context in PubMedQA often proved too expansive and not sufficiently focused. As a result, it was deemed unsuitable for our purposes and was not incorporated into our dataset.

D. MASH-QA

The MASH-QA dataset [14] is specifically designed for answering questions based on knowledge articles from the consumer health domain. It includes components such as questions, answers, and contexts. Given that the original context was often too extensive, we found the answers column, which includes all conceivable answers, to be more effective for our purposes. Therefore, we chose to use the answers as the new context for scoring. This strategy ensures the model encompasses every potential solution while focusing on essential and concise information. As a result, we have integrated this dataset into our project.

E. Dataset Preparation

The MASH-QA dataset was initially in JSON format, containing training, testing, and validation subsets without critical columns such as User Response, Score, and Feedback. To make the dataset more accessible and adaptable for our purposes, the first step involved converting the JSON files into a more manageable CSV format. The conversion process can be found in our code repository.

Next, we augmented the dataset with necessary columns using the Langchain Framework and OpenAI's GPT-3.5 Turbo model [2], simulating realistic user interactions and scoring responses.

To achieve this, we employed a prompt-based setup that generates two types of answers for each question—Answer A (good, complete) and Answer B (average, incomplete). This method ensures a diversity of answer quality, significantly enriching the training dataset with varied response scenarios . Responses were evaluated using a scoring system that reflects accuracy and relevance, with feedback for improvements, making this augmented data crucial for training our model to assess context-based answers. Documentation of this augmentation process is detailed in our code repository.

III. Model Training

In our project, our primary focus was on learning and implementing autoregressive decoder-only models. We conducted experiments using two key models: the LLaMA 2 7B model from Meta [12] and the Mistral 7B model from Mistral AI [5]. These models were selected based on their performance on benchmark tasks and their compatibility with the GPU memory available for training. Due to limited computational resources, we were unable to work with larger models.

We adopted a structured prompt format inspired by the Alpaca instruction format [11], encompassing key elements such as "Context", "Question", "Answer", "Score", and "Feedback" during the training phase. This format allowed us to systematically input user data, assess the model's responses, assign scores, and provide constructive feedback. During inference, we utilized a streamlined format comprising "Context", "Question", and "Answer" to ensure efficient querying and generation of responses.

We explored both instruction-tuned and non-instruction-tuned variants of these models to assess their effectiveness in our specific context. One of our objectives was to understand the impact of instruction tuning on model performance and adaptability. We used instruction-tuned variant to generate completions for our structured prompt, aiming to train the model to provide appropriate scores and feedback. We framed the prompt differently for the instruction-tuned variant. This approach allowed us to explore different strategies for enhancing the model's learning and performance.

Although we intended to explore distributed training techniques such as PyTorch Distributed Data Parallel (DDP) [9] and Fully Sharded Data Parallel (FSDP) [13], we were constrained to a single GPU machine. This limitation prevented us from leveraging the scalability benefits offered by distributed training methods.

To enhance model performance and efficiency within our

resource constraints, we employed Parameter Efficient Fine Tuning Techniques such as Low-Rank Adaptation (LoRA) [4] and Quantized Low-Rank Adaptation (QLoRA) [3]. These techniques enabled us to optimize model parameters effectively while mitigating the computational demands associated with training larger models.

IV. Experimental Setup

We conducted a series of experiments aimed at optimizing our model for efficient training and memory usage. The primary focus was on quantization and configuration adjustments to achieve a balance between model accuracy and resource constraints.

- **Quantization Exploration**: We experimented with different quantization levels to reduce memory usage. Initially, we attempted 8-bit quantization, but due to memory limitations, we further reduced it to 4 bits.
- **LoRA Configurations**: After extensive testing, we opted for utilizing query, key, value, gate, and output projections in our model. We set the LoRA rank to 32, LoRA alpha to 16, and dropout to 0.05.
- **Training Parameters**: We fine-tuned our model for 3 epochs, saving checkpoints every 500 steps. Train loss was logged every 20 steps for monitoring. Batch size was 8 with gradient accumulation of 2.
- **Optimization and Constraints**: We employed the paged AdamW optimizer [10] [7] with a learning rate of 2e-4 and weight decay of 0.001 for efficient convergence. Context length was limited to 2048 tokens due to GPU memory constraints, although the model architecture supported larger context lengths.
- **Sampling Strategies**: We conducted an extensive exploration of various hyperparameters, specifically focusing on temperature, top-k, and top-p values, to optimize the performance of our model.
  For the temperature parameter, we found that setting it to 0.9 produced optimal outcomes. Regarding the top-k parameter, we determined that a value of 40 was most effective. Finally, we observed that setting the top-p (nucleus) parameter to 0.9 provided superior results. By combining a temperature of 0.9, a top-k value of 40, and a top-p setting of 0.9, we achieved the best performance in terms of generating high-quality and diverse outputs.
- **Hardware and Training Time**: Our final model was trained on an Nvidia P100 16 GB GPU and took approximately 36 hours to converge, considering the complexities of the chosen configurations and training settings.

This approach ensured that our model achieved a balance between memory efficiency, training stability, and performance, resulting in an optimized solution for our project's objectives.

A. Pipeline

Our project incorporates a comprehensive pipeline Figure 1 comprising multiple stages to process user input effectively. The initial input includes the context, question, and answer provided by the user, which is subsequently formatted into our predefined instruction template. This formatted data is then fed into our model for analysis and evaluation. The model processes this input and generates an output score along with constructive feedback tailored to enhance user experience and performance.

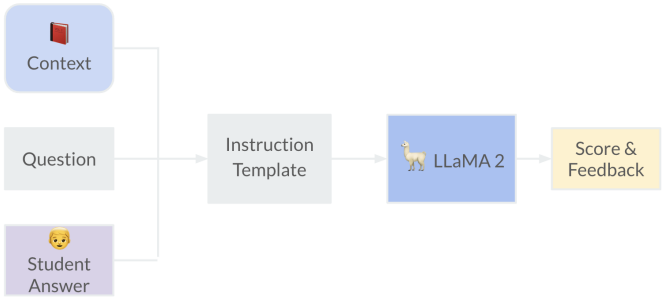

Fig. 1. Pipeline

V. Results

The project successfully implemented a machine learning-based approach to enhance the assessment of student answers. The model training process involved fine-tuning the Mistral 7B and the LLaMA 2 7B pre-trained Large Language Models (LLMs) on the dataset for 3 epochs.

The dataset consists of questions, contexts, two sets of answers, corresponding scores, and feedback for each question-context-answer pair. The model's predictions of scores and feedback were compared with the ground truth scores and feedback provided in the dataset.

For each question-context-answer, the model generates a score and feedback based on its learned parameters. After comparing the model's predictions with the ground truth scores and feedback, it was observed that the model's predictions were very close to the ground truth. This indicates that the model performed well in assessing student answers. In table I, we compare the output of our model with the ground truth result.

Comparison of Model Output with Ground Truth Results

Question	What are some tips to stay healthy during dialysis?
Context	Hemodialysis patients are also at an increased risk for infections.
Answer	Check your access site daily for redness, pus, and swelling. If you see any, call your doctor. Keep the bandage that covers your catheter clean and dry.
Ground Truth Score	10
Ground Truth Feedback	Perfect answer, well done!
Model Score	10
Model Feedback	Correct and concise answer.

TABLE I

This close alignment between the model's predictions and the ground truth scores and feedback underscores the effectiveness of the model in accurately evaluating student answers. It suggests that the model has successfully learned to capture the essential patterns and features from the training data, enabling it to make reliable assessments of student responses.

To further validate the model's performance, we conducted extensive testing with multiple scenarios, as demonstrated in Figure 2. The user interface showcases the generated score for the student's provided answer. Through these experiments, we observed that the model consistently assigns higher scores to answers that closely match the expected responses based on the given context. Conversely, when the student's answer is completely irrelevant to the context, the model appropriately assigns a very low score. This behavior highlights the model's ability to effectively discern the relevance and quality of student answers in relation to the provided context.

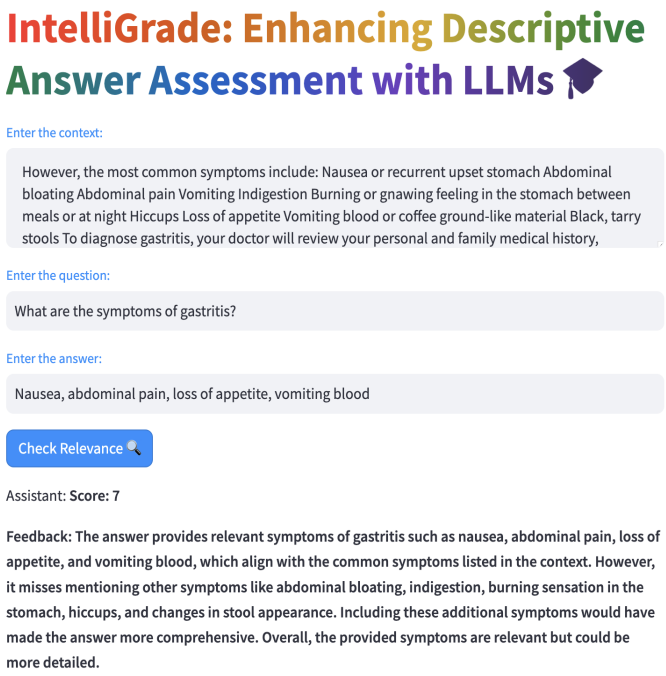

Fig. 2. Streamlit User Interface with score (7)

The project also focused on enhancing usability for educators through the integration of a user-friendly interface 2 using Streamlit. The integration with Streamlit can allow educators to interact with the model seamlessly, providing an intuitive interface for inputting student answers and receiving instant feedback. Educators can easily input questions, contexts, and student answers into the interface, and the model would generate scores and feedback in real-time.

VI. Conclusion

Intelligrader is quite significant in the development of educational evaluation where it makes use of large language models and helps in transforming the process through which we assess and provide feedback to the student answers for a given question in the provided context. By automating the grading process and delivering personalized, relevant feedback, Intelligrade empowers educators to focus on what truly matters: inspiring and guiding their students to reach their full potential. The results we obtained speak for themselves, since Intelligrade would produce scores that basically emulate human graders and provide insightful data in the form of feedback to promote student development. This marks an important milestone in our journey to come up with an innovative system that improves education experience for both students and graders.

Our approach, which makes use of the robust LLaMA 2 7B model and incorporates advanced techniques such as adaptive fine-tuning, dynamic feedback generation, and contextual answer evaluation, has produced a reliable solution that can reliably evaluate student replies for a variety of topics and question kinds. The ease of use and possibility for seamless integration with existing educational platforms augment Intelligrade's impact and accessibility. We are thrilled about the potential of AI-driven assessment to transform education and assist student success on a never-before-seen scale as we consider our accomplishments and look forward to future improvements.

VII. Future Work

As we look ahead, we can see that there is a great deal of room for improvement and growth in our project. Therefore, to improve our model's performance and usefulness, we intend to investigate a number of strategies, such as expanding the training dataset, looking into alternative pre-trained language models, and adding sophisticated features like support for multiple correct answers and thorough evaluation feedback. By connecting our system with well-known learning management systems and creating dynamic user interfaces for a more engaging experience, we also want to expedite the assessment process. Expanding the model's adaptability for domain-specific educational applications will increase its versatility even more.

VIII. Link to GitHub Repository

Here is the link to the GitHub repository of our project - https://github.com/vishant-mehta/fai-project

IX. Individual Contributions

A. Akshat Gandhi

I explored and analyzed datasets, developing code for data cleaning and augmentation.

B. Vishant K. Mehta

I was responsible for dataset exploration and analysis and fine-tuning the Mistral 7B model.

C. Aswath Senthil Kumar

I developed the user interface for our project using Streamlit and explored the MS Marco dataset.

D. Amrit Gupta

I worked on the training scripts, model selection, and experimental setup for the project.

Acknowledgment

We would like to express our sincere gratitude to Dr. Rajagopal Venkatesaramani for his invaluable guidance and support throughout this project. We are also thankful to the Khoury College of Computer Sciences and Northeastern University for providing the resources necessary for its completion.

References

[1] Payal Bajaj et al. *MS MARCO: A Human Generated MAchine Reading COmprehension Dataset*. 2018. arXiv: 1611.09268 [cs.CL].

[2] Tom Brown et al. "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[3] Tim Dettmers et al. "QLoRA: Efficient Finetuning of Quantized LLMs". In: *arXiv preprint arXiv:2305.14314* (2023).

[4] Edward J Hu et al. "LoRA: Low-Rank Adaptation of Large Language Models". In: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=nZeVKeeFYf9.

[5] Albert Q. Jiang et al. *Mistral 7B*. 2023. arXiv: 2310.06825 [cs.CL].

[6] Qiao Jin et al. "PubMedQA: A Dataset for Biomedical Research Question Answering". 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)*. 2019, pp. 2567–2577.

[7] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1412.6980.

[8] Tom Kwiatkowski et al. "Natural Questions: a Benchmark for Question Answering Research". In: *Transactions of the Association of Computational Linguistics* (2019).

[9] Shen Li et al. *PyTorch Distributed: Experiences on Accelerating Data Parallel Training*. 2020. arXiv: 2006.15704 [cs.DC].

[10] Ilya Loshchilov and Frank Hutter. *Decoupled Weight Decay Regularization*. 2019. arXiv: 1711.05101 [cs.LG].

[11] Rohan Taori et al. *Stanford Alpaca: An Instruction-following LLaMA model*. https://github.com/tatsu-lab/stanford_alpaca. 2023.

[12] Hugo Touvron et al. *Llama 2: Open Foundation and Fine-Tuned Chat Models*. 2023. arXiv: 2307.09288 [cs.CL].

[13] Yanli Zhao et al. *PyTorch FSDP: Experiences on Scaling Fully Sharded Data Parallel*. 2023. arXiv: 2304.11277 [cs.DC].

[14] Ming Zhu et al. *Question Answering with Long Multiple-Span Answers*. Available: https://people.cs.vt.edu/mingzhu/papers/conf/emnlp2020.pdf. 2020.