Automated Grading and Feedback System for Assignments using NLP

Neel Singh
MS CS
Purdue Fort Wayne
singn03@pfw.edu

Ajinkya More
MS CS
Purdue Fort Wayne
morea04@pfw.edu

Arjun Malik
MS CS
Purdue Fort Wayne
malia01@pfw.edu

Abstract

This work addresses the issue of manual grading inefficiencies in college departments, which often lead to delayed feedback and hinder the learning process. The proposed solution is the creation of an automated grading and feedback system powered by Large Language Models (LLMs). These sophisticated models, with their advanced natural language processing capabilities, offer a quick and accurate way to grade assignments, circumventing the time-consuming nature of traditional grading methods. This not only expedites the feedback students receive, enhancing their learning experience but also lightens instructors' workload, enabling them to concentrate on more personalized and impact teaching methods. The ultimate goal of this automated system is to improve the overall efficiency of assessment workflows in college departments, fostering a more dynamic and responsive educational environment.

1 Introduction

This paper introduces an Automated Grading and Feedback System utilizing advanced Natural Language Processing (NLP) techniques to address the challenges of manual grading in academic departments. The system aims to overcome delays in feedback, enhance the learning experience for students, and reduce the workload on instructors by leveraging Large Language Models (LLMs). The study outlines the creation of a custom dataset, tackles data imbalance, and experiments with various modeling techniques, ultimately identifying a superior Retrieval Augmented Generation model. The results highlight the system's potential to revolutionize the grading process in educational institutions by providing swift and accurate feedback through advanced NLP capabilities.

1.1 Motivation

The motivation behind this project stems from the need to enhance the learning experience for students and reduce the burden on instructors within our college department. Manual grading often leads to delayed feedback, which can hinder students' progress and cause anxiety. By automating the grading process, we can provide students with rapid and constructive feedback, ultimately improving their learning outcomes. Moreover, the Automated Grading and Feedback System aligns with the broader educational objectives, fostering a more supportive and efficient learning environment. This innovative solution not only addresses immediate challenges but also contributes to the overarching goal of advancing educational practices within our institution.

1.2 Related Work

Various works have been done to automate the process of evaluation. However, they are mostly focused upon using classifiers to assign grades to the student's assignment(Mieskes and Padó, 2018). Whereas other researchers apply state-of-the-art techniques like Reinforcement learning only to assign scores to the essay. (Wang et al., 2018) Our approach combines both the objectives that is score prediction and feedback generation. The score prediction task further narrows down the objective and goes beyond classification strategies to assign grades to assignments. Further, our approach incorporates feedback generation which is a fundamental part of a holistic evaluation system. Our approach also enables us to evaluate both coding as well as textual assignments. Lastly, our work focuses on developing an automated evaluation system for Purdue University Fort Wayne, a first in itself.

2 Model/Algorithm to Address Problem

The system will consist of several key components:

2.1 Dataset

Our carefully curated dataset, sourced from real-time student information, comprises 21 assignments examples. It includes homework details, task descriptions, student code, reports, and grade points, with manually crafted reference feedback for each assignment. To reduce data imbalance, we have introduced negative examples by creating synthetic data. We took the regular assignments and deleted portions of them to make it incomplete and assigned a score to them reduced in proportion to the amount of content deleted. The structured dataset, with columns such as Homework and Task Solution Code, provides a holistic view of student performance. Our primary focus is on accurate score prediction and feedback generation, ensuring a comprehensive and adaptable solution for automating the grading and feedback process.

2.2 Methodology

Our final approach involved using Retrieval Augmented Generation with the help of FAISS and Palm2 to produce best results. A step by step approach is mentioned below which describes our plan of action.

1. **Collect Data** - This includes homework tasks, code files, report files, homework expectations, scores, and feedback.

2. **Dealing with data imbalance** - Most assignments are scored > 40. This forces the model to memorize only from "good" assignments. To even this out, we synthetically generate "poor" assignments by providing partial/irrelevant/mixed codes and reports for the given task and assigning them a lower score.

3. **Dividing the problem** - Evaluation depends on 2 tasks: giving a Score and providing Feedback. Score prediction is more of a regression task, whereas providing feedback is a text generation task.

4. **Establishing baseline** - There are two parts of modelling. First we assess the baseline performance by applying various models to the given task. Here we treat score prediction and feedback classification as different tasks and thus different models are fit for regression and classification. Second, once the baseline is established we integrate both tasks by eliciting response(score prediction and feedback generation) from LLM models. Note, after integration of tasks all the data will be fed together to the model during training. This is different than process adopted before integration where we pursued both tasks as separate modelling problems.

5. **Data preprocessing** - The data preprocessing involves converting textual columns into vectorized format. Whereas for feedback generation, we undertook multiple experiments such as using model's default tokenizer with padding, truncating excess tokens, changing max length of model response,varying temperature, stopword removal and augmenting instructions with data for instruction tuning.

6. **Integrating score and feedback generation** - By feeding data through contexts into pretrained LLM. While quality of feedback was assesed based on BleU score, the score predictions were evaluated using mean squared error(MSE). After getting the integrated response (score and feedback) from our LLM models, we extracted the score(prediction) and compared it with actual scores given for that assignment.

7. **Vectorization** - Using FAISS vector database to reduce large data context into smaller document chunks to bypass limits to context length and Palm2 to speed up retrieval of relevant documents for faster vector similarity search.

8. **Inference** - User inputs the homework task along with their code and report which, together with predefined instruction, becomes the query. The pretrained LLM (Palm2) assists the response (score and feedback) generation through a question-answering chain and is aided by efficient similarity search.

2.3 Experimentation

We experimented with various modeling techniques, each with different strengths and drawbacks. Traditional ML models are simple and less prone to overfitting, but limit the depth of evaluation. Fine-tuning works well with large datasets but can overfit and produce false results with smaller ones. Context-based prompting is reliable but limited by context length. Retrieval Augmented Generation is promising and addresses context limitations but struggles with lengthy queries. We use metrics

Model	Mean Squared Error(Score Prediciton)
Linear Regression	0.48
Ridge Regression	1.12
Random Forest Regression	0.89
Support Vector Regressor	1.0
K-Nearest Neighbors Regressor	0.5
Deep Neural Network(Layer1(32 units), Layer2(64 units))	1.2

Table 1: Standalone Score prediction results(before integration)

Model	F1 score(Feedback multiclass classification)
Logistic Regression	0.81
Random Forest Classifier	0.92
XGBoost classifier	0.95

Table 2: Standalone Feedback classification results(before integration)

like Mean Square Error and BLEU score to evaluate each method, highlighting the need to choose the most suitable model based on task requirements and dataset characteristics.

2.4 Modelling

In our quest to enhance the process of feedback generation, we delved into testing various models, including Deepset/tinyRoberta (Chan et al., 2023), google/FlanT5xxl (Chung et al., 2022), and Google/Palm2. Among these, the Google/Palm2 (Kasliwal, 2023) model stood out, outperforming all others in our evaluations. This model was pretrained on question answering, code evaluations and multilingual language translation tasks. This reflected in its superior performance over other models, thus making it our top choice for feedback generation. We adopted Retrieval Augmented Generation considering various factors such as small size of our dataset, its susceptibility to overfiting during finetuning,the need to prevent hallucinations and bias and to avoid issues of limited context length. Therefore we augmented the model with FAISS(Facebook AI similarity search) vector database which enabled faster retrieval of results. With Langchain's text splitter and FAISS we were able to break down data into smaller documents which were converted into embeddings. While FAISS (Johnson et al., 2019) narrowed model's search space by providing most relevant document embeddings, the model(Palm2) conducted efficient similarity search between the query and document embeddings to find the most relevant response through a question answering chain. Since the model already identified features of good and poor homework examples when the data was fed, it is able to provide the grade along with appropriate feedback for the student's homework.After getting the response (score and feedback) from our LLM model, we extracted the score(prediction) and compared it with actual scores given for that assignment. Loss was calculated through Mean squared error.

3 Measurement of Success

For evaluating feedback generation we are relying on BleU score, which compares model response with reference feedback(manually generated) and assigns a score between 0 and 1 based on the extent of n-grams overlap between the two texts(higher is better). We have used BleU score from NLTK(Natural Language Toolkit) which considers all possible combinations of n-grams overlap (unigram,bigram till n-grams) and averages them out before giving out the final score. After getting the integrated response (score and feedback) from our LLM models, we extracted the score(prediction) and compared it with actual scores given for that assignment. Loss was calculated through Mean squared error.

4 Results

The Retrieval Augmented Generation model incorporating Palm2 and FAISS vector database exhibits superior performance compared to other models, achieving a commendable balance between Mean Square Error and BleU Score. Notably, Fine Tuning and Instruction Tuning, despite introducing innovative approaches, displayed lower BleU scores, highlighting potential challenges in generating re-

Modeling Approach	Pros	Cons	Assessment
Fine Tuning(Gpt2)	Works well with large, labeled datasets	Leads to overfitting and hallucinations	BLEU score: 0.11, prompt-like responses
Instruction Tuning(Deepset/tiny-roberta)	Theoretically works well with small datasets	Similar results to fine-tuning with overfitting	BLEU score: 0.09, prompt-like responses
Context-based Prompting (PALM2)	Reliable results, reduced hallucinations	Limits to context length, decreasing accuracy	BLEU score: 0.53, reliable within limits
Retrieval Augmented Generation (PALM 2)	Reliable results, solves limited context	FAISS embeddings fit for our data size	BLEU score: 0.65, mostly reliable

Table 3: Integrated Score-Feedback generation

sponses that closely align with human assessments. Context-Based Prompting yields a moderate BleU Score, indicating reliable performance contingent on the length of context provided.

These findings result from a dual evaluation framework encompassing both score predictions and feedback generation. The generated feedback consistently aligns with homework expectations, offering comprehensive evaluations of content and specific details on incomplete sections. However, it's noteworthy that the predicted scores often surpass the expected actual scores. This discrepancy may arise from the dataset's tendency to assign liberal scores to assignments, and even negative examples (those with deleted content) should ideally receive even lower scores than currently assigned.

5 Additional Analysis

The overall results indicate that transfer learning that makes use of pretrained models trained on huge data, can show very promising results with finetuning. For finetuning, in most of the cases we require labelled datasets which can now be generated to a significant extent with the help of LLMs themselves. Though the quality of this synthetic data may not be as good as original data, yet they should be enough enable the model to perform decently well on the given task. This approach can take automated evaluation to another pedestal.

6 Future Scope

The future of the Automated Grading and Feedback System using NLP is promising. We aim to make it more robust by augmenting data (generating synthetic data with help of LLMs), further fine-tuning and using multiple metrics for evaluation. We also plan to integrate it with Learning Management Systems, broaden its scope, and introduce personalized education through adaptive learning features. We're focusing on continuous refinement, inclusion of various grading styles, and improved user accessibility. Moreover, we're eager to address ethical implications, aiming for a secure, bias-minimized, and all-encompassing educational tool.

7 Conclusion

As we conclude, we are optimistic about the potential that transfer learning and Generative AI hold in making high-quality automated evaluations, an integral part of Learning Management System. The prospect of making both students' and teachers' lives easier by accelerating the grading process and providing prompt feedback is genuinely exciting. The dedication demonstrated in meticulously creating a custom dataset and testing various models shows a deep commitment to this system's effectiveness. Although we've encountered some challenges, recognizing the importance of having an appropriately sized dataset and selecting the most fitting model shows a sensible approach to improving the system in the future. In essence, this research signifies a positive move towards a more streamlined and efficient grading process not only in our university but also in other schools and colleges.

References

Branden Chan, Timo Möller, Malte Pietsch, Tanay Soni, and Michel Bartels. 2023. deepset/tinyroberta-squad2. *Proceedings of ACL.*

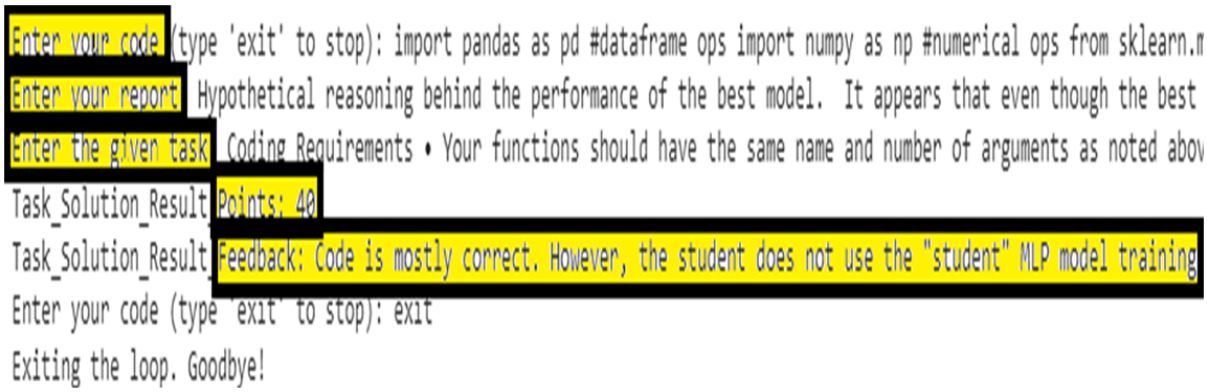

Figure 1: Autograder Demo

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Nirant Kasliwal. 2023. Hugging face datasets: dbpedia-entities-google-palm-gemini-embedding-001-100k. https://huggingface.co/datasets/nirantk/dbpedia-entities-google-palm-gemini-embedding-001-100K.

Margot Mieskes and Ulrike Padó. 2018. Work smart - reducing effort in short-answer grading. In *Proceedings of the 7th workshop on NLP for Computer Assisted Language Learning*, pages 57–68, Stockholm, Sweden. LiU Electronic Press.

Yucheng Wang, Zhongyu Wei, Yaqian Zhou, and Xuanjing Huang. 2018. Automatic essay scoring incorporating rating schema via reinforcement learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 791–797, Brussels, Belgium. Association for Computational Linguistics.

1

[1]Github repository: https://github.com/arjunmalik11/AutoGrader