Automated Assignment Evaluator Using Retrieval-Augmented Generation

Prof. Sushmita Choure[1] Raj Yogesh Mohite[2] Shritej Santosh Waikar[3] Shravani Prakash Pande[4] Gayatri Mahadev Mudkanna[5]

[1,2,3,4,5]Department of Artificial Intelligence and Machine Learning

[1,2,3,4,5]Zeal Polytechnic, Pune, Maharashtra, India

Abstract — Automated assignment evaluation can reduce the time required to assess student answers, but evaluating answers written in different words remains a challenge. This paper proposes an Automated Assignment Evaluator using Retrieval-Augmented Generation (RAG). The proposed system retrieves relevant information from approved study materials and supplies it to a large language model along with the question, reference answer, and marking rubric. The model then generates a suggested score and feedback based on the retrieved evidence and the rubric. The workflow includes answer extraction, document retrieval, rubric-based evaluation, feedback generation, and teacher review. The system is intended to support educators rather than replace their final judgement. Its performance should be assessed by comparing its scores with human-assigned marks and measuring scoring error, agreement, evaluation time, and feedback usefulness. As the system is proposed, experimental results are not claimed in this paper.

Keywords: Retrieval-Augmented Generation, Automated Assignment Evaluation, Large Language Model, Semantic Similarity, Rubric-Based Grading, Educational Technology

I. Introduction

Assignment evaluation is an important part of education because it helps teachers measure student understanding and provide feedback. Manual checking can require considerable time, especially when a class has many students. Simple automated methods can compare exact words, but they may fail when a student expresses a correct idea using different wording. Large Language Models (LLMs) can interpret natural-language answers, while Retrieval-Augmented Generation (RAG) supplies relevant information from an external knowledge source. Combining retrieval with a marking rubric may help an evaluator ground its decisions in course material and explain the marks it suggests. This paper presents a proposed RAG-based system for evaluating written assignments and generating feedback for teacher review.

II. Problem Statement and Objectives

Existing manual evaluation takes time, while basic keyword-matching systems may not recognize semantically correct answers that use different words. LLM-only evaluation may also produce unsupported judgements or apply marking criteria inconsistently. The problem addressed is to design a system that retrieves relevant course material and applies a predefined rubric to produce more traceable suggested marks and feedback. The objectives are to:

1) accept student answer in a digital document;
2) retrieve relevant content from approved study resources;
3) evaluate answers against a reference answer and rubric;
4) generate understandable feedback; and
5) allow a teacher to verify or edit the suggested result.

III. Literature Review

Lewis et al. introduced Retrieval-Augmented Generation, combining a language model with a retriever that accesses external documents. Their work showed that retrieval can improve the specificity and factual grounding of generated responses on knowledge-intensive tasks [1]. This provides the technical foundation for retrieving relevant educational material before evaluating an answer. Aggarwal et al. surveyed automatic short-answer grading methods, including traditional machine-learning and deep-learning approaches, and discussed challenges such as feedback, multilingual support, and evaluation datasets [2]. Their review helps frame automated grading as more than simple keyword matching. Haldar et al. proposed RUBRA, a short-answer grading system that combines LLMs and RAG with retrieved domain context and rubric-based assessment [3]. This is closely related to the present project and shows that RAG-assisted grading is an existing research direction. Therefore, the proposed project should be evaluated through a clearly defined implementation and measured performance rather than claiming that RAG-based grading is entirely new.

IV. Proposed System and Methodology

The proposed system follows a sequence of processing stages. First, the student submits an assignment as text or a supported document. The system extracts and separates the answers according to the questions. Next, the question and relevant terms are used to search an indexed collection of approved notes, textbooks, or reference material. A retrieval component returns the most relevant passages. The retrieved passages, question, reference answer, student answer, and marking rubric are then provided to an LLM through a structured prompt. The model is instructed to assess each rubric criterion, suggest a score within the allowed range, and provide concise feedback supported by the available context. The output includes the suggested marks, feedback, and the retrieved evidence used for assessment. Finally, a teacher reviews the output and can correct the score before it is finalized.

V. System Architecture

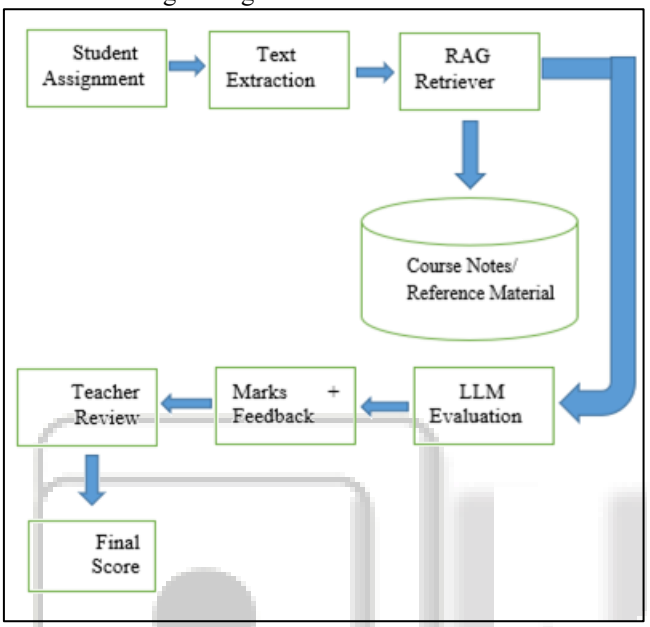
The main modules are:

1) Assignment Input, which receives student submissions;
2) Text Extraction, which converts supported documents into text;
3) Knowledge Base, which stores approved learning resources;
4) Retriever, which finds relevant passages using embeddings and similarity search;
5) RAG Evaluation Engine, which combines retrieved context with the rubric and sends it to the LLM;

All rights reserved by www.ijsrd.com

6) Feedback and Score Generator, which formats the suggested marks and comments; and
7) Teacher Review, which enables human verification.

Example: if the question asks for a definition of Artificial Intelligence, the reference answer may state that AI enables machines to perform tasks that normally require human intelligence. A student may write that AI allows computers to do tasks that usually need human thinking. The evaluator should consider the meaning and the rubric, rather than requiring an exact word-for-word match. The example illustrates the intended behaviour; actual accuracy must be measured during testing.



VI. Technologies and Tools

Python can be used for the application logic and text processing. A document-reading library can extract text from supported files. An embedding model can represent passages and answers as vectors, and a vector database or similarity-search library can retrieve relevant study content. An LLM performs rubric-guided evaluation and generates feedback. A simple web interface may be developed using Streamlit or Flask. The final tool choices will depend on available hardware, cost, privacy requirements, and project constraints.

VII. Evaluation Plan and Discussion

Because this paper describes a proposed system, no experimental performance results are claimed. After implementation, the system should be tested on a permitted set of sample answers with marks assigned independently by a teacher or qualified evaluator. The same questions and rubric should be used for human and automated scoring. Mean Absolute Error (MAE) can measure the average absolute difference between system scores and human scores. Score agreement can show how consistently the system matches human evaluation. Evaluation time can be recorded for both manual and automated workflows, and feedback can be reviewed for relevance, correctness, and clarity. A useful comparison is an LLM-only baseline versus the RAG-based system, to test whether retrieval improves scoring and feedback. The dataset, rubric, model, prompts, and test procedure should be documented so that the findings can be reproduced.

VIII. Advantages and Limitations

Potential advantages include reduced repetitive checking, feedback delivered more quickly, consideration of meaning rather than exact wording, and traceability through retrieved course material. A consistent rubric can also make the evaluation process easier to inspect. Limitations include incorrect retrieval, incomplete study materials, model-generated errors, bias, sensitivity to prompts, and variation in human grading. The system may also struggle with diagrams, handwritten responses, calculations, or answers requiring nuanced judgement unless these are specifically supported. For these reasons, the system should provide suggested marks and retain teacher oversight instead of making high-stakes decisions independently.

IX. Conclusion and Future Scope

This paper proposes an Automated Assignment Evaluator using RAG to support rubric-based scoring and feedback. The system retrieves relevant educational content and combines it with the question, reference answer, and marking criteria before generating a suggested evaluation. The approach aims to make feedback more understandable and to support teachers in reducing repetitive work. Its effectiveness remains to be established through experiments against human-assigned scores.

Acknowledgment

The authors thank their project guide, Sushmita Choure, and the Department of Artificial Intelligence and Machine Learning for guidance and support during the planning of this project.

References

[1] P. Lewis et al., “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” Advances in Neural Information Processing Systems, vol. 33, pp. 9459–9474, 2020.
[2] D. Aggarwal, P. Bhattacharyya, and B. Raman, “Automatic Short Answer Grading: A Survey,” IIT Bombay, 2024.
[3] S. Haldar, S. Sengupta, and A. K. Das, “RUBRA: An Agentic AI System for Automatic Short Answer Grading Using LLMs and RAG,” 2025 International Conference on Computing, Intelligence, and Application (CIACON), 2025, doi: 10.1109/CIACON65473.2025.11189329.

All rights reserved by www.ijsrd.com