IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511156

Smart Answer Sheet Analysis and Personalized Feedback Using Retrieval-Augmented Generation

Prof. Sangeetha M[1], Srushti D[2], Nithin G[3], Ujwal Gh[4], Vinay V Gouda[5]

Dept. of Information Science and Engineering Vemana Institute of Technology Bengaluru, India[1-5]

Abstract: Marking handwritten, descriptive exam answers is still largely manual. An evaluator reads every script, judges whether the student really grasped the concept, and writes a mark. Two evaluators seldom agree exactly, and a single evaluator can drift during a long marking session. This paper presents SAGE (Smart Answer-sheet Grading Engine), a platform that combines optical character recognition, sentence embeddings, retrieval-augmented generation (RAG) and large-language-model (LLM) scoring in one workflow. An OCR pipeline digitizes scanned or handwritten scripts, the text is converted to dense vectors, and those are matched against an index built from reference and syllabus material. The LLM then scores each answer against the passages retrieved for it, rather than leaning only on what it remembers from training. We want to move past keyword matching and pattern classification, which earlier work finds brittle on descriptive answers, and arrive at scoring that can explain a mark in plain language. We cover the architecture, the scoring pipeline, the module design, the related work behind our choice of a retrieval-based ap-proach, and an evaluation protocol built on OCR accuracy, agreement with human graders and retrieval precision. The evaluation dataset has not been built and no experiments have been run, so the numbers reported here are targets, not measured results.

Keywords: automated answer evaluation, retrieval-augmented generation, optical character recognition, sentence embeddings, large language models, semantic similarity, educational technology, vector search.

I. INTRODUCTION

Most academic evaluation still relies on a person working through handwritten scripts one at a time. For short factual answers that is fine. For descriptive, concept-based answers it is slow, it differs from one evaluator to the next, and the student usu-ally gets back nothing but a number. Researchers have attacked automated grading from several directions: convolutional networks that match handwriting patterns, OCR paired with lexical measures such as TF-IDF, and, more recently, transformer-based sentence embeddings that compare meaning instead of wording. Each lifts some work off the evaluator. Yet none of them lets the system check its own score against verified subject content, and none explains a mark the way a human evaluator would.

SAGE grew out of that problem. Rather than asking a language model to grade from its training knowledge alone, the system first retrieves the reference and syllabus passages that bear on the question and only then requests a score. Our reason for using retrieval-augmented generation is practical. If the scor-ing step is grounded in retrieved text, the model is less likely to produce a convincing justification that nothing supports, and the system can point to the material a mark was based on.

The project brings together OCR to digitize handwritten scripts, sentence-embedding models to represent text, a vector database for retrieval, and an LLM for scoring and feedback, all delivered through a full-stack web application. Students, evaluators and administrators are the intended users, though this paper stays with the evaluation pipeline and leaves account management and institutional integration aside.

We make four contributions, all on the implementation side. First, we argue that retrieval-augmented generation, not pattern classification or lexical similarity, is the right way to ground automated scoring of descriptive answers, and we explain how it deals with weaknesses reported in earlier CNN- and TF-IDF-based systems. Second, we keep OCR digitization apart from semantic scoring so each can be evaluated on its own. Third, we define a modular pipeline where the retrieval index, the embedding model and the scoring model can each be swapped without redesigning the rest. Fourth, we lay out a reproducible evalua-tion protocol based on OCR character error rate, correlation with human evaluators and retrieval precision, and we keep target metrics clearly separate from measured results, since we have not run the experiments yet.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511156

II. RELATED WORK

A. Pattern-based and lexical approaches to answer grading

Shaikh et al. use a convolutional neural network that segments handwritten answer scripts and classifies the segments against learned handwriting and layout patterns [1]. That can automate the mechanical side of script processing, but it cannot judge whether the concept in a descriptive answer is correct. It recog-nizes shapes rather than meaning, and it breaks down quickly on open-ended free-text answers.

A second line of work pairs OCR with lexical similarity. One 2021 IEEE CCT study digitizes scripts with OCR and then scores each answer by TF-IDF cosine similarity against a model answer [2]. The approach is simple to build and quick to run, but accuracy is capped at both ends: by OCR quality on one side and by the limits of word overlap on the other. Two answers explaining the same concept in different words can receive very different scores, a familiar weakness of bag-of-words compari-son on descriptive text.

B. Embedding-based semantic scoring

More recent work swaps lexical overlap for semantic embeddings. A 2024 IEEE/Springer study grades exam answers using BERT-derived sentence embeddings, comparing meaning instead of exact wording [3]. This removes much of the brittleness of TF-IDF scoring. Even so, systems like this generally stop at a similarity score. They bring in no external verified content while scoring, and they give the student none of the concept-level feed-back needed to improve. SAGE builds on this embedding-based work and adds the two things it lacks: retrieval grounding against a knowledge base, and readable, generated feedback.

C. Sentence embeddings and transformer language models

The representation techniques SAGE relies on are well estab-lished. BERT introduced deep bidirectional transformer pre-training for language understanding [4]. Sentence-BERT then adapted the architecture with a siamese network, so that sentence embeddings can be compared efficiently using cosine similarity [5]. That is precisely what a retrieval-based grading pipeline needs for fast nearest-neighbor search over a reference corpus.

D. Retrieval-augmented generation for grounded outputs

We pair retrieval with a generative model because a language model's own parameters are neither a reliable nor a checkable source of facts. In grading this matters a lot, since a wrong or invented justification directly affects a student's mark. Retriev-ing relevant reference material before generation is intended to keep the output tied to what a verified knowledge base actually contains, not to what the model half-remembers from pretraining. It also lets the system show which passages a score rested on.

III. RESEARCH GAP AND CONTRIBUTIONS

The literature already offers strong individual pieces: CNN-based handwriting recognition, OCR with lexical scoring, and embedding-based semantic grading. What is missing is a single pipeline that combines OCR digitization, a retrieval step against verified reference content, and a generative feedback stage. CNN-and TF-IDF-based systems have no semantic grounding at all, and even the embedding-based system we reviewed ends at a similarity score with no retrieval and no explanation.

SAGE's contribution, then, is integration and grounding. We do not propose a new embedding model or a new language model. OCR, semantic retrieval and generative scoring stay as separate stages that can be tested independently, and retrieval is there mainly to keep the scoring model from drifting away from the syllabus or reference material that the answer should be judged against.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511156

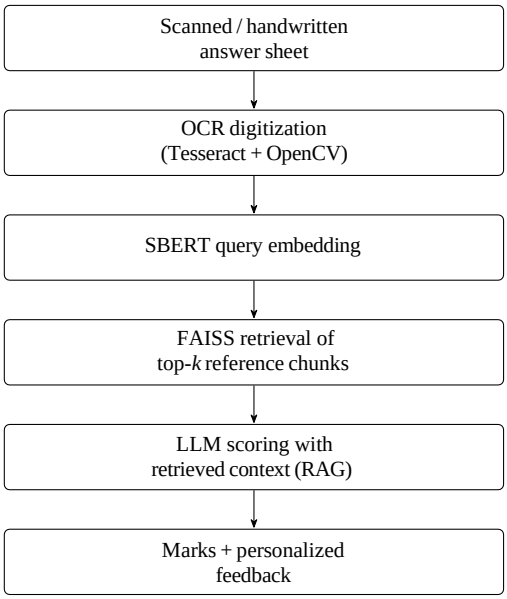


Fig. 1 Overall SAGE pipeline from script digitization to graded feedback.

IV. SYSTEM ARCHITECTURE

SAGE runs as a four-stage pipeline: image processing, natural language understanding, retrieval-based knowledge augmenta-tion, and AI-assisted feedback generation. The interface for stu-dents and evaluators is built with React.js/Next.js, and a FastAPI backend coordinates the OCR, embedding, retrieval and generation stages while storing application data in PostgreSQL. We plan to host the system on a cloud platform such as AWS or GCP.

We keep the stages apart because each one fails in its own way and is measured in its own way. OCR quality depends on handwriting legibility, scan resolution and preprocessing, and character error rate lets us measure it separately from grading. Retrieval quality depends on the embedding model and the ref-erence corpus, and standard information-retrieval metrics such as precision at k cover it. Scoring quality depends on how well the language model uses the retrieved context, which we can check through correlation with human-assigned marks. With the stages separated, a weakness can be traced to its source without working backwards from the final mark.

A. OCR and pre-processing

Handwritten or scanned answer sheets first go through OpenCV pre-processing (grayscale conversion, binarization, denoising and deskewing) and then through Tesseract OCR. The output is a block of machine-readable text for each answer, and the rest of the pipeline works from that block.

B. Semantic embedding and retrieval

A Sentence-BERT model turns each digitized answer into a dense vector. The same model indexes the reference and syllabus material in a vector store (FAISS or ChromaDB). Student answers and reference content therefore share one embedding space and can be compared by vector similarity rather than by surface text.

C. Retrieval-augmented scoring

For each student answer, the top-k most relevant reference chunks are pulled from the vector store and placed in the prompt for the large language model (GPT-4, LLaMA-3 or Mistral, de-pending on the deployment). The prompt also carries the grading rubric and the student's answer. We ask the model for a mark and a short justification that refers to the retrieved material, not only to its own internal knowledge. Because retrieval sits between the student's answer and the model's judgment, a score should be traceable to specific reference content instead of being an unconstrained model output.

D. Feedback generation

Besides the numeric mark, the LLM stage produces structured feedback: which concepts the student covered correctly, which were missing or wrong, and what to review. The student is meant to read this directly, not just the evaluator. It is the main reason we chose generation over plain classification.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511156

V. METHODOLOGY

We split the platform into three cooperating paths: digitization and representation, retrieval-grounded scoring, and the applica tion layer that shows results to students and evaluators.

A. Digitization and representation path

An answer script enters the system as a scanned image or PDF. OCR converts it to text, and the embedding stage turns that text into a vector. We evaluate this path separately from grading, with OCR character error rate as the main metric. If the digitized text is wrong, judging the grading quality that follows would be pointless.

B. Retrieval-grounded scoring path

Once an answer is digitized, the system runs semantic retrieval before it calls the scoring model. We do not have the model grade unaided and check the result afterwards. This ordering is the main design choice separating SAGE from the embedding-only and lexical-only systems in Section II. The retrieved context is part of the input to scoring, not an explanation attached after the fact.

C. Application path

The application layer (React.js/Next.js, FastAPI, PostgreSQL) gives students and evaluators access to the pipeline and stores scripts, scores and feedback as records. This is ordinary web-application engineering and not a research contribution in itself. We include it because a grading pipeline that evaluators cannot use and students cannot see has little practical value, however good its scoring.

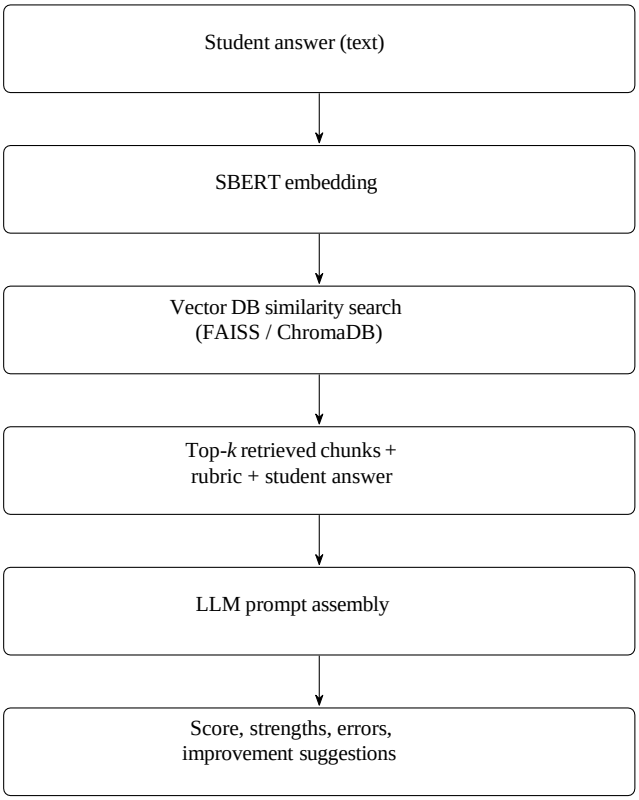


Fig. 2 Retrieval-augmented scoring path: retrieval happens before generation, not after.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511156

TABLE I TARGET METRICS AND CURRENT EVIDENCE STATUS

Stage	Target metric	Evidence status
OCR	CER ≤ 10% (i.e., ≥90% accuracy) on clean hand-writing	From the project brief only; not yet evaluated
Retrieval	Precision@5 ≥ 80%	From the project brief only; not yet evaluated
Grading	Correlation with human score ≥ 0.85 (Pearson *r*)	From the project brief only; not yet evaluated
Throughput	≤ 30 sec/sheet end-to-end	From the project brief only; not yet evaluated

VI. IMPLEMENTATION

The project specification calls for Tesseract OCR and OpenCV for digitization, Sentence-BERT for embeddings, FAISS or ChromaDB for vector storage and retrieval, and a large language model (GPT-4, LLaMA-3 or Mistral) for scoring and feedback. These are served through a React.js/Next.js and FastAPI/PostgreSQL stack on AWS or GCP.

Full reproducibility would also require exact library versions, prompt templates, rubric formats, the reference corpus used for retrieval, and the deployment configuration. We do not report these here, and we chose not to guess at model version numbers, corpus sizes, prompt text, latency figures or infrastructure details that we cannot back up.

VII. EVALUATION METHODOLOGY

A. Evaluation principles

Since the system has three fairly independent stages, we plan to evaluate each on its own as well as giving an end-to-end

TABLE II DESIGN-LEVEL CONSIDERATIONS AND EVIDENCE STATUS

Concern	Design intent	Evidence status
Answer-sheet confidentiality	Role-based access, so only the enrolled student and the assigned evaluator can see a script	Design-level; not independently verified
Retrieval-corpus integrity	Reference material is curated and version-controlled, so retrieved context can be traced	Design-level; not independently verified
Model output reliability	Retrieved context lowers, but does not remove, the chance of an ungrounded score or justification; a human evaluator can always override	Design-level; robustness testing not done

score. OCR will be measured with character error rate (CER) and retrieval with precision at *k*. Grading will be measured by agreement with human evaluators: Pearson's *r* suits continuous marks, while Cohen's κ is a better fit if marks are grouped into ordinal grade bands. One overall number can hide systematic disagreement on particular question types or mark ranges, so we also plan per-question and per-band breakdowns instead of a single correlation figure.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511156

B. Target metrics and evidence status

The figures in Table I are design targets taken from the project brief, not measured outcomes. We do not yet have a finished repository, reference corpus, pilot answer-sheet dataset or evalua-tion logs, so Table I shows what we plan to measure, not numbers we have collected.

VIII. SECURITY AND DATA HANDLING CONSIDERATIONS

A system that handles student work and personal academic records raises access-control and data-integrity issues apart from the grading logic. Three apply to SAGE (Table II): confiden-tiality of answer sheets, integrity of the retrieval corpus, and reliability of model output. None has been verified against a running implementation yet.

The retrieval-augmented design lowers the risk of an incor-rect or fabricated score but does not remove it. For that reason we describe the platform as producing a mark and feedback that an evaluator can review, not a final grade that nobody can question.

IX. EXPERIMENTAL SETUP

A full evaluation of SAGE would test each stage against ground truth. OCR output would be compared with manually transcribed scripts to obtain CER. Retrieval output would be compared with a labeled set of question-to-passage mappings to obtain precision@*k*. Final scores would be compared with a panel of human evaluators who grade the same answers independently, so that agreement among the humans can be set beside agreement between the model and the humans.

We do not yet have the implementation repository, reference corpus, pilot dataset or evaluator scoring logs this would need, so this section and Table I describe the protocol only.

X. RESULTS

We have no measured OCR error rate, retrieval precision, grad-ing correlation or processing time to report, so Table I lists them as targets, not estimates. The strongest honest claim at this stage concerns the design. It shows how OCR, semantic retrieval and generative scoring can be combined so that each stage can be tested independently and the final score rests on retrieved refer-ence content, not on the language model alone. Any claim about grading accuracy or efficiency must wait for real experiments.

XI. DISCUSSION

The key design decision in SAGE is where retrieval sits. Placing retrieval before generation, rather than letting a language model score. OCR will be measured with character error rate (CER) and retrieval with precision at k. Grading will be measured by agreement with human evaluators: Pearson's r suits continuous marks, while Cohen's κ is a better fit if marks are grouped into ordinal grade bands. One overall number can hide systematic disagreement on particular question types or mark ranges, so we also plan per-question and per-band breakdowns instead of a single correlation figure.

This choice does not remove the basic risk of LLM-based scoring: fluent text from a model is not the same as a correct evaluation. Retrieval narrows what the model draws on, but grading quality still rests on three things. The reference corpus has to be accurate and complete. The retrieval step has to surface material that is actually relevant. And the model has to use that material instead of falling back on what it already "knows." That is why the protocol in Section VII measures retrieval quality and grading correlation separately, instead of folding them into one accuracy number.

Keeping OCR, retrieval and generation as separate stages has a practical benefit too. The OCR engine, the embedding model or the LLM can each be replaced as better tools appear, with-out redesigning the rest of the pipeline, provided the interfaces between stages stay stable.

XII. LIMITATIONS

The biggest limitation of this paper is missing evidence, not the architecture. The implementation repository, the retrieval corpus, a pilot dataset of scanned answer sheets and evaluator scoring logs are not available for this version, so none of the targets in Table I has been verified.

 This work is licensed under a Creative Commons Attribution 4.0 International License



International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026
DOI: 10.17148/IJARCCE.2026.1511156

OCR accuracy depends heavily on handwriting quality, scan resolution and language or script. The 90–95% accuracy target applies to clean handwriting, not to every submission a real deployment would receive. Retrieval quality depends entirely on how complete and correct the reference corpus is, and a gap in the corpus would not appear as an obvious retrieval failure; it would appear as subtly wrong grounding for the LLM's score. Finally, the LLM scoring stage has not been checked against a panel of human evaluators, so SAGE should be read as a prototype-stage design, not a validated grading system.

XIII. FUTURE WORK

Next, we want to build and annotate a pilot dataset of scanned answer sheets with independent human grading so that we can run the correlation analysis. We also want to extend the reference corpus to more subjects and exam formats, test OCR across a wider range of handwriting styles and scan qualities, and add explainability features that show an evaluator exactly which retrieved passages contributed to a score. Multilingual OCR and retrieval are further goals, as is a proper evaluator-in-the-loop override workflow, so that a corrected mark can be fed back into the system.

XIV. CONCLUSION

SAGE takes a retrieval-augmented approach to evaluating de-scriptive answer scripts. OCR digitization, semantic embedding, retrieval against a verified reference corpus and LLM scoring are treated as separate, cooperating stages rather than one opaque model. The main contribution is the retrieval-before-generation design, which is meant to ground the model's scoring and feedback in verifiable reference content and to fill a specific gap in earlier CNN- and TF-IDF-based systems, which have no seman-tic grounding in verified subject material. Because we do not yet have the repository, reference corpus or pilot evaluation logs, this paper reports target metrics and evidence status, not measured results. A follow-up empirical paper can build on it directly by replacing each evidence-status entry with measurements from the actual implementation and a documented pilot dataset.

ACKNOWLEDGMENT

The authors thank Mrs. Sangeetha, Assistant Professor, Depart-ment of Information Science and Engineering, for her guidance during the development of the SAGE project.

REFERENCES

[1]. E. Shaikh, I. Mohiuddin, A. Manzoor, G. Latif, and N. Moham-mad, "Automated Grading for Handwritten Answer Sheets using Convolutional Neural Networks," in *Proc. IEEE Conf.*, 2019.
[2]. "Automated Evaluation of Subjective Handwritten Answers using OCR and TF-IDF Cosine Similarity," in *Proc. IEEE CCT Conf.*, 2021.
[3]. "Automated Answer Evaluation System using Machine Learning and Semantic Similarity," IEEE/Springer, 2024.
[4]. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Under-standing," in *Proc. NAACL-HLT*, 2019.
[5]. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in *Proc. EMNLP*, 2019.

 This work is licensed under a Creative Commons Attribution 4.0 International License