iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** X **Month of publication:** October 2026

DOI: https://doi.org/10.22214/ijraset.2026.85533

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
AISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue X Oct 2026- Available at www.ijraset.com

AnswerKey Validation Intelligence: AI- Assisted Exam Evaluation Framework

Mrs. Dhana Lakshmi R[1], Vishal Suriya K R[2], Thirumurugan S[3], Thilip T[4]
[1]*Assistant Professor (Project Guide),* [2, 3, 4]*B.E. Students (Reg. Nos. 6176AC23UCS165, 6176AC23UCS166, 6176AC23UCS174), Department of Computer Science and Engineering, Adhiyamaan College of Engineering (An Autonomous Institution), Dr. M.G.R. Nagar, Hosur – 635130, Tamil Nadu, India*

Abstract: ***Manual assessment of descriptive examination answers requires substantial faculty time and can produce variation when different evaluators interpret the same response. This paper presents AnswerKey Validation Intelligence, a web-based system designed to assist—not replace—faculty in processing and evaluating descriptive answer sheets. The proposed workflow combines optical character recognition (OCR), natural language processing (NLP), semantic comparison with faculty-provided model answers, keyword-based evidence, and human verification. Students submit answer sheets as images or PDFs; the system extracts text and organizes responses for question-level comparison. Preliminary marks are generated from configurable evaluation criteria and presented to faculty for correction and approval before publication. The platform also supports examination management, role-based access, record storage, and student result viewing. The design draws on prior work in automatic short-answer grading, contextual language representations, sentence embeddings, and transformer-based OCR. The project report describes the application workflow and its main modules, but does not provide a sufficiently documented benchmark dataset or numerical grading-accuracy measurements. Accordingly, this paper focuses on the system design, implementation workflow, and a reproducible evaluation plan rather than claiming unmeasured performance gains. A key design principle is that OCR errors and semantic scores must remain reviewable so that final marks are approved by a faculty member.***

Keywords: ***automatic answer evaluation, optical character recognition, TrOCR, natural language processing, semantic similarity, keyword matching, human-in-the-loop assessment.***

I. INTRODUCTION

Descriptive questions are commonly used to assess whether students can explain concepts, connect ideas, and communicate knowledge in their own words. Evaluating these answers manually is demanding when a course has many students or when examinations contain several descriptive questions. An evaluator must read each response, compare it with the expected concepts, decide how partial credit should be awarded, and record the result. The work is not limited to matching a student's wording with a model answer: a correct explanation may use different terms, sentence structures, or examples while expressing the same underlying idea.

Computer-assisted grading offers a way to reduce repetitive work, but reliable assessment remains challenging. Exact keyword matching can miss valid paraphrases and may award credit to a response that includes the expected terms without explaining them correctly. Conversely, semantic similarity alone may overestimate an answer that shares broad vocabulary with the reference but omits a required concept. A further complication is the quality of the source document. Handwritten answer sheets can contain unusual character shapes, crossed-out words, uneven spacing, and low-contrast scans, all of which may affect OCR output.

AnswerKey Validation Intelligence addresses these challenges through a staged workflow. OCR is used to convert uploaded answer sheets into editable text; question-level responses are compared with faculty-defined model answers and keywords; preliminary scores are produced; and faculty members inspect the extracted text and marks before publishing results. The system also links examination information, submissions, and results within a web application. Its intended role is decision support: the final assessment remains under academic staff control.

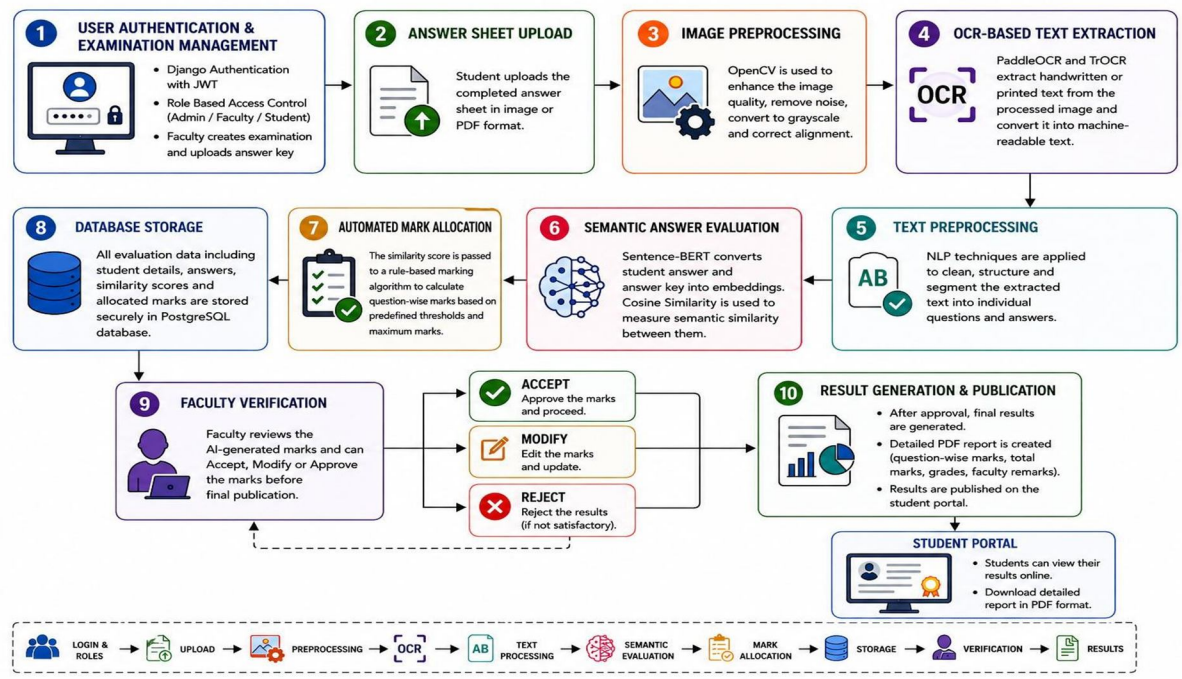
A. Objectives and Contributions

- Develop a unified workflow for examination setup, answer-key preparation, student submission, assisted evaluation, faculty verification, and result publication.
- Use OCR to reduce manual transcription of answer sheets and retain the extracted text for review.

©IJRASET: All Rights are Reserved

- Combine semantic comparison and keyword evidence so that evaluation is not restricted to exact word matches.
- Maintain role-based access and organized records for examinations, submissions, preliminary marks, and approved results.
- Identify a transparent evaluation procedure that can be tested against faculty-assigned marks before making claims about accuracy or efficiency.

The principal contribution is the integration of these stages into a faculty-supervised assessment workflow. This paper does not claim that the system has already demonstrated superior grading accuracy; that conclusion requires a controlled experiment using representative answer scripts and independent reference marks.



II. RELATED WORK

Automatic short-answer grading has developed through information retrieval, linguistic analysis, machine learning, and semantic approaches. Burrows, Gurevych, and Stein surveyed the history and evaluation of automatic short-answer grading systems and emphasized that comparison across systems requires clear task definitions and evaluation methods [1]. This is relevant to the present project because the quality of an answer-scoring system cannot be established only by describing its features; the grading task, reference data, and scoring measures must also be reported.

Contextual language models provide representations that use information from surrounding words rather than treating each term in isolation. Devlin et al. introduced BERT, a bidirectional Transformer model that learns contextual representations and can be adapted to language-understanding tasks [2]. Reimers and Gurevych proposed Sentence-BERT, which produces sentence embeddings suitable for efficient similarity comparison [3]. Such representations can support comparison between a student's response and a model answer when their wording differs. However, a similarity score is not equivalent to correctness: it should be interpreted alongside the expected concepts, question context, and marking rubric.

OCR is another essential part of the workflow because the grading stage depends on the text extracted from the uploaded script. Smith's overview of the Tesseract OCR engine describes key components of a widely used OCR system for document text recognition [4]. Li et al. introduced TrOCR, a Transformer-based recognition framework that combines image and text Transformers and reports results on printed and handwritten text-recognition tasks [5]. These approaches illustrate the development of document recognition, but their published performance cannot be assumed to apply directly to the project's answer sheets. Script quality, handwriting style, language, layout, and preprocessing may change recognition outcomes. The project combines these strands into a practical application: document extraction, semantic and keyword-based comparison, and a human review stage. The distinction between prior work and this implementation is important. The cited papers establish the relevance of OCR and semantic representations; they do not validate the accuracy of this particular application. A project-specific benchmark is therefore required.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

III. PROPOSED SYSTEM AND ARCHITECTURE

The system is organized around three user roles: administrator, faculty member, and student. The administrator manages access and system-level records. Faculty members create examinations, enter model answers, specify key concepts or keywords, and review preliminary evaluation results. Students submit answer sheets and view results after faculty approval. Role-based permissions are intended to prevent users from accessing functions outside their responsibilities.

The processing pipeline consists of six main stages. First, the faculty member creates an examination and defines its questions, model answers, keywords, and maximum marks. Second, the student uploads a supported image or PDF of the answer sheet. Third, the OCR component extracts text, with preprocessing applied where needed to improve readability. Fourth, extracted responses are associated with the corresponding questions. Fifth, the evaluation component compares each response with the model answer and configured keywords to produce a preliminary score. Finally, the faculty member reviews the extracted text and score, corrects errors or marks, and approves the result for publication.

A. Main Modules

- Authentication and role management: validates user credentials and restricts access according to role.
- Examination and answer-key management: stores examination details, questions, reference answers, keywords, and maximum marks.
- Answer-sheet upload: accepts student submissions and associates each file with the relevant examination and student.
- OCR and text review: extracts machine-readable text from the uploaded document and provides an opportunity to correct recognition errors.
- AI-assisted evaluation: compares the extracted response with the model answer and checks for relevant keywords or concepts.
- Faculty verification and results: allows staff to adjust preliminary marks, approve the final score, and publish results.
- Database and reporting: maintains user, examination, submission, evaluation, and result records for later retrieval.

B. Data Flow

The input to the system is an examination record, a faculty-approved answer key, and a student answer-sheet file. The OCR stage produces extracted text, which is segmented or mapped to individual questions. The evaluation stage uses the corresponding model answer and marking criteria to estimate a preliminary score.

Both the extracted text and the score are retained for review. Faculty verification is a required checkpoint before results are visible to students. If the OCR output is incorrect or the preliminary score is inappropriate, the evaluator can edit the relevant field and save the final decision. The architecture therefore preserves a traceable distinction between machine-generated suggestions and approved academic results.

Suggested figure for the final manuscript: insert a simple flow diagram showing Faculty Setup → Student Upload → Image/PDF Preprocessing → OCR Text Extraction → Question-wise Matching → Preliminary Marks → Faculty Review → Published Results. The figure should be generated from the actual implemented workflow rather than copied from another publication.

IV. METHODOLOGY

A. Document Input and OCR

A submitted answer sheet may be supplied as an image or a PDF. The document is checked for an accepted format and passed to the text-extraction stage. Depending on the file type, the pages may need to be rendered as images before recognition. Image preprocessing can include grayscale conversion, noise reduction, contrast adjustment, deskewing, and cropping. These operations should be applied only where they improve legibility; excessive processing can remove strokes or alter characters. The project report identifies TrOCR and PaddleOCR among the considered OCR technologies. The final manuscript should name the exact engine and model version used in the tested build, rather than implying that every listed tool was integrated if some were only explored.

The OCR output is not treated as error-free ground truth. Recognition mistakes can change technical terms, numbers, negation, or mathematical notation and thereby affect grading. The application therefore makes extracted text available for human inspection and correction. In a future quantitative evaluation, OCR quality should be measured against manually transcribed text using character error rate (CER) and word error rate (WER).

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

B. Response Matching and Preliminary Scoring

For each question, the student's extracted response is compared with the faculty-provided model answer. Semantic similarity can be computed from sentence embeddings, for example using a Sentence-BERT model, and cosine similarity can be used to measure the alignment between the two embeddings. For vectors u and v, cosine similarity is defined as:

$cos(u, v) = (u \cdot v) / (||u|| \, ||v||)$

The similarity value provides a signal of semantic relatedness, not a direct probability that an answer is correct. The scoring stage also checks the presence of faculty-specified keywords or concepts. The system may combine these signals according to a rubric configured for the question. A generic formulation is:

$S = \alpha M + \beta K$, *where* $\alpha + \beta = 1$

Here, M denotes a normalized semantic-matching score, K denotes a normalized keyword/concept score, and α and β are weights chosen during system configuration. This expression is a design template, not a claim that a specific formula or set of weights has already been validated. The implementation must document the actual formula, thresholds, model name, and normalization method used. The preliminary score is then mapped to the question's maximum marks and presented to faculty for review.

A robust rubric should account for required concepts, partial credit, irrelevant content, contradictions, and the possibility that a response is correct without containing the exact expected keywords. The system should not rely on a single similarity threshold for every subject or question type. The marking criteria should be established before evaluation and kept consistent across submissions for the same question.

C. Human Verification and Data Management

The faculty review stage is central to the design. Evaluators can inspect OCR text, compare the response with the question and model answer, adjust the preliminary score, and approve the final result. The database should retain, at minimum, the submission identifier, extracted response, preliminary score, final score, review status, and relevant timestamps. Maintaining these values separately supports auditability and makes it possible to analyse where human corrections are frequent.

The project report describes Django as the web framework and Django REST Framework as the API layer, with PostgreSQL intended for structured storage. The final version must clearly distinguish the technology actually used in the tested deployment from alternatives or planned components. Security measures should include password hashing, role-based authorization, server-side validation, restricted file uploads, and protection against unauthorized access to examination records. Student answer sheets contain educational records and should be handled according to institutional privacy requirements.

V. IMPLEMENTATION AND EVALUATION PLAN

The application is described as a web-based platform with separate interfaces for administrators, faculty, and students. Its core workflow connects authentication, examination setup, answer-sheet upload, OCR, assisted evaluation, faculty verification, database storage, and result publication. The project report identifies Python and Django as the main application technologies and describes OCR and semantic analysis as the central AI components. Before publication, the authors should confirm the exact package versions, database configuration, OCR engine, semantic model, and deployment settings used in the version being evaluated.

A. Functional Validation

Functional testing should verify each stage independently and then test the complete workflow. Authentication tests should confirm that each role can access only its permitted pages and actions. Upload tests should include valid image and PDF files, unsupported extensions, empty files, oversized files, and unreadable scans. OCR tests should confirm that extracted text is associated with the correct student and examination. Evaluation tests should verify that the correct answer key and maximum marks are used, and that faculty edits persist after saving. Result-publication tests should ensure that students see only their own approved results.

A reproducible report should provide the number of test cases, pass/fail counts, software and hardware environment, and any known defects. The project documentation describes the intended workflow and modules, but does not provide a complete, independently reproducible quantitative benchmark for grading quality. Consequently, system functionality and grading accuracy should be reported as separate dimensions.

B. Quantitative Evaluation Protocol

To measure grading performance, prepare a consented and appropriately anonymized dataset of student responses covering multiple questions and answer styles.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

At least one qualified faculty member should assign reference marks using a fixed rubric; a second evaluator is desirable for measuring inter-rater variation. The same answer scripts should then be processed by the system without revealing the reference marks to the model. OCR quality and score agreement should be analysed separately.

- OCR: report CER and WER by document type and handwriting quality.
- Mark agreement: report mean absolute error (MAE) and root mean squared error (RMSE) between system preliminary marks and faculty reference marks.
- Ranking/association: report Pearson or Spearman correlation where appropriate, alongside the number of responses evaluated.
- Efficiency: compare the time taken for manual marking and AI-assisted marking under the same conditions, including faculty review time.
- Human correction: report how often faculty members edit extracted text or change the preliminary score.

MAE can be computed as the mean absolute difference between predicted marks and reference marks; RMSE gives greater weight to larger errors. Results should be broken down by question type and answer-sheet quality where the sample size permits. Any exclusions, missing values, model thresholds, and rubric choices must be documented. No accuracy percentage, time-saving percentage, or statistical significance should be reported unless it has been calculated from actual observations.

VI. RESULTS AND DISCUSSION

The available project documentation describes the end-to-end workflow and the intended functions of the application: role-based access, examination and answer-key management, image/PDF submission, OCR extraction, semantic and keyword-based preliminary evaluation, faculty correction, and result publication. These functions form the basis for a functional demonstration of the proposed approach. However, the report does not provide enough measured data to establish a numerical grading-accuracy rate or to conclude that the system outperforms manual evaluation.

The design has several practical strengths. It organizes assessment tasks in a single platform and keeps the preliminary score separate from the faculty-approved mark. Semantic comparison may help identify answers that express expected concepts with different wording, while keyword evidence can make selected concepts explicit in the marking process. Editable OCR output provides a way to correct recognition errors before the final score is confirmed. These are design benefits; their actual effect on accuracy and workload must be measured.

There are also important limitations. OCR performance may vary with handwriting, page layout, image quality, language, and mathematical notation. Semantic similarity can confuse topical similarity with correctness and may fail to recognize a critical misconception. Keyword matching can miss valid paraphrases or reward isolated terms. Scores can also depend on the quality of the faculty-provided model answer and the consistency of the rubric. For these reasons, fully automatic publication of marks is not recommended. Human review, traceable edits, and question-level validation should remain part of the workflow.

Table 1 should be completed after the planned experiment. It is intentionally presented as a reporting template rather than filled with invented values.

Measure	What it evaluates	Value
CER / WER	OCR transcription errors	To be measured
MAE / RMSE	Difference from faculty marks	To be measured
Correlation	Agreement in score variation	To be measured
Review time	Manual vs. assisted marking time	To be measured
Correction rate	OCR or score edits by faculty	To be measured

VII. CONCLUSION AND FUTURE WORK

This paper presented AnswerKey Validation Intelligence, a faculty-supervised system for supporting descriptive answer evaluation. The workflow combines OCR-based text extraction, question-wise comparison with model answers, semantic similarity, keyword evidence, faculty verification, and structured result management. The system is intended to reduce repetitive handling of answer sheets while retaining human control over final academic decisions. Its practical value depends on the quality of OCR output, the appropriateness of the marking rubric, and the reliability of the semantic scoring method.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The current project documentation establishes the system concept and workflow but does not provide sufficient quantitative evidence to support claims of improved grading accuracy or measured time savings. The next step is to conduct a controlled evaluation with anonymized answer scripts and faculty-assigned reference marks, report OCR and grading metrics, and analyze the cases in which the system disagrees with evaluators. Further development may include question-specific rubrics, better response segmentation, support for additional languages and notation, calibrated partial-credit rules, audit logs, and feedback to help faculty identify uncertain predictions. The system should be treated as an assessment assistant rather than an autonomous authority on student performance.

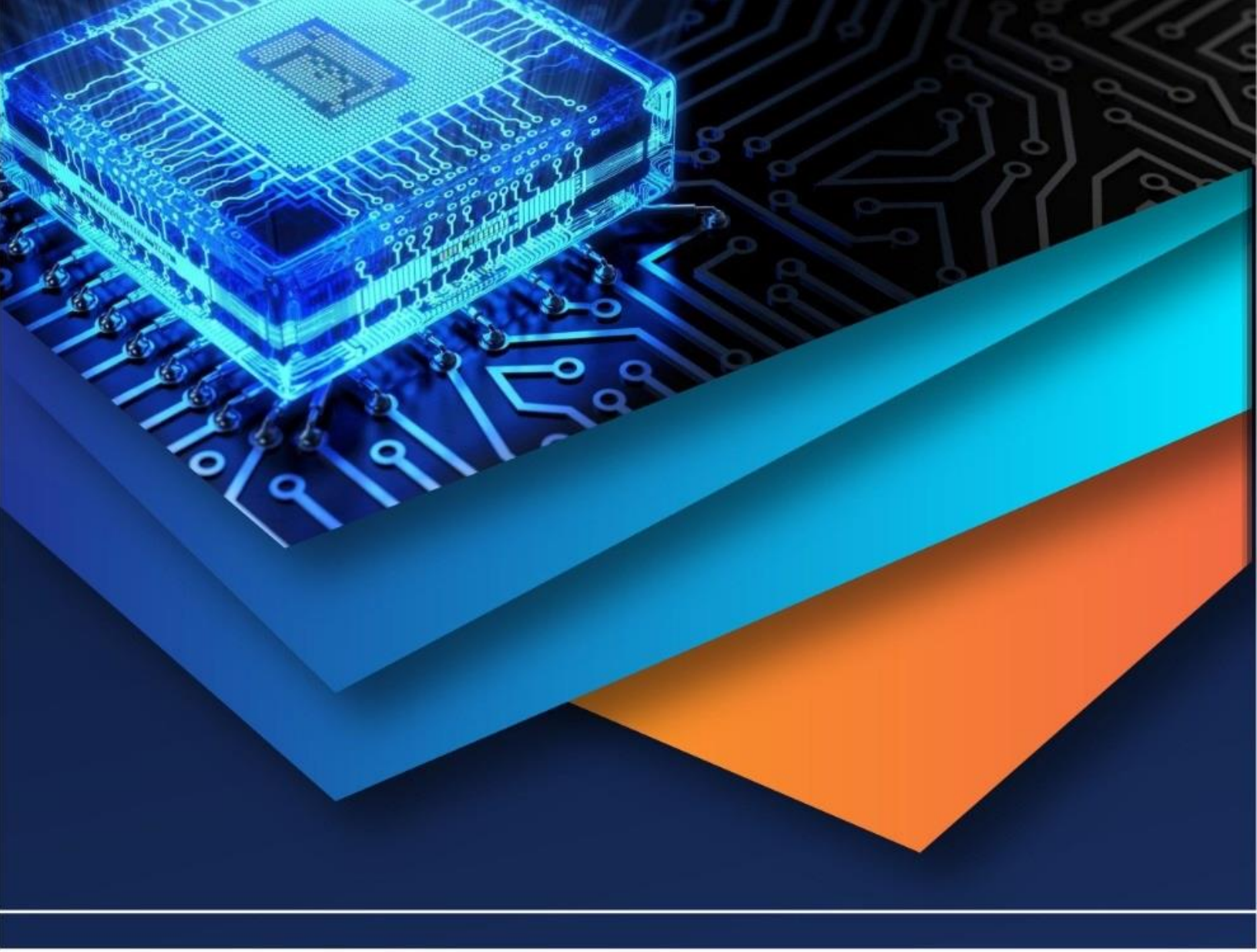
VIII. ACKNOWLEDGMENT

The authors thank Dr. R. RADHAKRISHNAN, Principal, and Dr. G. FATHIMA, Professor and Head, Department of Computer Science and Engineering, Adhiyamaan College of Engineering, Hosur, for the opportunity and encouragement, and the project coordinator Mrs. DHANA LAKSHMI R AP\CSE and faculty of the department for their support.

REFERENCES

[1] S. Burrows, I. Gurevych, and B. Stein, “The Eras and Trends of Automatic Short Answer Grading,” International Journal of Artificial Intelligence in Education, vol. 25, pp. 60–117, 2015, doi: 10.1007/s40593-014-0026-8.
[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Lnguage Understanding,” in Proceedings of NAACL-HLT, vol. 1, pp. 4171–4186, 2019, doi: 10.18653/v1/N19-1423.
[3] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” in Proceedings of EMNLP-IJCNLP, pp. 3982–3992, 2019, doi: 10.18653/v1/D19-1410.
[4] R. Smith, “An Overview of the Tesseract OCR Engine,” in Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR), pp. 629–633, 2007, doi: 10.1109/ICDAR.2007.4376991.
[5] M. Li et al., “TrOCR: Transformer-Based Optical Character Recognition with Pre-trained Models,” Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 11, pp. 13094–13102, 2023, doi: 10.1609/aaai.v37i11.26538.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)