H V Poojitha, 2026, 14:5
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Automated Evaluation of Academic Assignments Using Ai

H V Poojitha[1], Harshitha C[2], Madhulekha M[3] , Likhitha B[4]
Department Of Artificial Intelligence And Data Science
Gsss Institute Of Engineering And Technology For Women, Vtu, Mysuru, Karnataka, India

Abstract— Evaluating academic assignments manually requires considerable time and effort, particularly when an instructor must assess many student submissions against common questions and evaluation criteria. Differences in interpretation may also lead to variation in the feedback provided to students. This paper presents an AI-assisted academic assignment evaluation system that supports teachers in reviewing written student responses. The system accepts questions, reference answers, and evaluation rubrics from a teacher, extracts text from submitted assignment files and compares student responses with the corresponding reference material. A pretrained Sentence-BERT model is used to generate semantic sentence representations, and cosine similarity is applied to estimate the semantic closeness between the student's response and the reference answer. The resulting similarity information is considered within a rubric-oriented scoring workflow, while rule-based checks support the evaluation process. The application is implemented using a React/Vite frontend, a Python FastAPI backend, and an SQLite database. Sentence-BERT is used in it pretrained form without task-specific fine-tuning. The completed testing reported for this project focuses on functional verification of the application workflow; a benchmark against human-assigned marks and quantitative grading-accuracy results have not been established. The system is therefore intended as a decision-support tool that can reduce repetitive evaluation effort and provide structured feedback, with the teacher retaining responsibility for final academic judgement.

Keywords— Artificial Intelligence, Automated Assignment Evaluation, Natural Language Processing, Sentence-BERT, Semantic Similarity, Cosine Similarity, Rubric-Based Assessment, Educational Technology.

I. INTRODUCTION

Academic assignments are widely used to assess students' understanding, subject knowledge, and ability to explain concepts. In conventional evaluation, teachers read each submission, compare it with the expected answer, and assign marks according to the assessment criteria. Although this approach allows instructors to consider context and reasoning, it becomes time-consuming when the number of students or questions increases. Repeated evaluation of similar answers can also create workloads and may result in differences in scoring or feedback.

Artificial intelligence and natural language processing (NLP) provide methods for processing and comparing written responses. Earlier automated essay-scoring systems used linguistic features and statistical methods, while later approaches employed machine-learning and neural-network models to learn patterns associated with scores [1]– [5]. For short answers, semantic similarity methods have been explored to compare a response with one or more reference answers [6], [7]. These developments provide a basis for systems that assist teachers with routine evaluation tasks.

A major difficulty in automated evaluation is that a correct answer may be expressed using words or sentence structures different from those in the reference answer. Exact keyword matching may fail to recognize such responses even when their meaning is similar. Contextual language representations help address this limitation by representing sentences according to their semantic content. Sentence-BERT (SBERT) produces sentence embeddings that can be compared efficiently using cosine similarity [8].

This paper presents an AI-assisted system for evaluating academic assignments. A teacher supplies the question,

© 2026 H V Poojitha, This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly credited.

reference answer, and rubric-oriented evaluation criteria. Students submit their answers as assignment files. The application extracts the text, compares the response with the reference answer using pretrained Sentence-BERT embeddings, and applies a rubric-oriented scoring workflow to support the teacher's assessment. The system is designed to assist rather than replace the teacher.

The main contributions of this work are: (i) an integrated web-based workflow for teacher-provided questions, reference answers, rubrics and student submissions; (ii) semantic response comparison using pretrained Sentence-BERT and cosine similarity; (iii) a structured evaluation process supported by rule-based checks; and (iv) a functional application implementation using React/Vite, FastAPI and SQLite. Since the completed project testing was functional rather than a controlled grading-accuracy study, no numerical accuracy or human-agreement claim is made.

II. LITERATURE SURVEY

Research on automated assessment has developed from feature-based essay scoring to neural and transformer-based language processing. Page's early work explored the possibility of computer-based essay grading [1]. Landauer, Foltz and Laham introduced latent semantic analysis as a method for representing contextual meaning in text, which later influenced automated evaluation approaches [2]. Attali and Burstein described the e-rater system, which uses linguistic features and scoring models to support essay assessment [3].

Subsequent studies examined machine-learning methods for automated essay scoring. Phandi et al. investigated domain adaptation for essay scoring [4], while Taghipour and Ng proposed a neural approach using recurrent networks [5]. Alikaniotis et al. used neural networks to model text quality and predict scores [6]. Hussein et al. reviewed automated language essay-scoring systems and discussed both handcrafted-feature and automatically learned-feature approaches [7]. These studies show the progression from manually designed indicators to learned language representations.

Automatic short-answer grading is closely related to assignment evaluation because it compares a student's response with expected content. Mohler and Mihalcea studied text-to-text semantic similarity for automatic short-answer grading [8].

Burrows et al. reviewed the development and components of automatic short-answer grading systems, identifying preprocessing, similarity measures, and evaluation as important stages [9]. Lubis et al. explored semantic similarity based on word embeddings for short-answer grading [10]. Ormerod et al. presented an approach combining neural networks and latent semantic analysis for scoring short constructed responses [11].

Transformer-based language models have further improved contextual text representation. Vaswani et al. introduced the Transformer architecture based on attention mechanisms [12]. Devlin et al. presented BERT, which learns bidirectional contextual representations [13]. Reimers and Gurevich introduced Sentence-BERT, adapting BERT into sentence embeddings suitable for semantic similarity comparisons [14].

The present project uses a pretrained Sentence-BERT model without additional fine-tuning and applies cosine similarity to compare student answers with reference answers. Recent surveys continue to identify explainability, reliability, fairness and meaningful feedback as important challenges in automated assessment [15], [16]. Research on automated scoring also highlights the importance of suitable datasets and evaluation against human scores [17], [18].

In contrast to studies that train and quantitatively benchmark scoring models, the current project is presented as a functional application prototype. Its contribution is the integration of document text extraction, semantic comparison, rubric-oriented evaluation, and teacher-facing workflow in one application. A controlled comparison with human grading remains in future work.

III. PROPOSED METHODOLOGY

The proposed system follows a sequence of teacher configuration, student submission, text extraction, semantic comparison, rubric-oriented evaluation and result presentation. The teacher defines the assessment information before student responses are processed. The backend coordinates document handling and semantic analysis, while the database stores the application records.

1. *Teacher Input and Assessment Setup*

The teacher provides the assignment question, reference answer, and rubric or evaluation criteria. The reference answer represents the expected subject content, while the rubric specifies the aspects that should be considered during evaluation. These inputs form the basis for comparing and reviewing student responses. The system is intended to use the teacher-defined material rather than assume that one fixed answer is the only acceptable wording.

2. *Student Assignment Submission*

Students submit their completed assignments through the application. The uploaded file is passed to the backend for text extraction. The extracted content is used for subsequent processing. The quality of the extracted text depends on the source document; scanned files or poorly formatted documents may require additional optical character recognition or manual correction. OCR capability is not claimed as an implemented feature unless confirmed in the final project materials.

IV. TEXT PREPROCESSING AND RESPONSE PREPARATION

The extracted text is organized so that the relevant answer can be compared with the corresponding question and reference answer. Basic text preparation is performed to make the content suitable for semantic processing. The system must preserve the meaning of the student's response; preprocessing should therefore avoid removing technical terms, important negations, or meaningful expressions.

V. SENTENCE-BERT-BASED SEMANTIC REPRESENTATION

Sentence-BERT is used to convert the student response and reference answer into dense numerical vectors, commonly called sentence embeddings. Unlike direct string matching, these representations are intended to capture semantic information in the sentences. The project uses a pretrained Sentence-BERT model and does not fine-tune it on a custom assignment dataset.

Let the embedding of the student response be represented by A and the embedding of the reference answer by B. Their semantic similarity is calculated using cosine similarity:

1. Similarity (A, B) = (A · B) / (||A|| × ||B||)

The cosine similarity value indicates the directional similarity between the two embedding vectors. A higher value generally indicates greater semantic closeness. It should not be interpreted directly as a guaranteed mark or as proof that the answer is academically correct. The value is used as an input to the rubric-oriented evaluation workflow.

2. *Rubric-Oriented Scoring And Rule-Based Evaluation*

The system considers semantic comparisons together with the teacher-provided rubric and rule-based evaluation logic. This makes the assessment more structured than relying only on a single similarity value. The rubric provides the criteria against which the answer is reviewed, while rule-based checks support consistent handling of the evaluation steps. Exact rubric weights, threshold values and score aggregation formulas are not specified in the project details available for this draft; they are therefore not invented here and should be inserted from the implementation or literature-survey material if documented.

3. *Feedback And Result Presentation*

After processing, the application presents the evaluation output to support the teacher's review. The intended workflow includes a score-oriented result and feedback related to the response and assessment criteria. The teacher remains responsible for reviewing the result, correcting unsuitable evaluations, and

finalizing marks. This human-in-the-loop arrangement is important because semantic similarity alone cannot fully judge factual correctness, reasoning quality, originality, or the completeness of every subject-specific answer.

4. *System Architecture And Technology Stack*

The application follows a frontend–backend architecture. The user interface is developed using React with Vite. The backend services are implemented in Python using FastAPI. SQLite is used for persistent application data. The backend handles incoming requests, assignment text processing, and semantic evaluation. The Sentence-BERT model is invoked for embedding generation, and cosine similarity is calculated for comparing the response and reference answer.

Layer / Component	Technology	Role
Frontend	React / Vite	Teacher and student interaction; input and result display.
Backend	Python / FastAPI	Request handling, file processing and evaluation workflow.
Semantic model	Pretrained Sentence-BERT	Generation of sentence embeddings.
Similarity measure	Cosine similarity	Estimation of semantic closeness between answers.
Database	SQLite	Storage of application data and records.

5. *Overall Workflow*

The overall workflow is summarized as follows: (1) the teacher creates an assessment by entering a question, reference answer and rubric; (2) the student uploads an assignment; (3) the backend extracts the response text; (4) the text is prepare d for semantic processing; (5) Sentence-BERT generates embeddings for the student and reference answers; (6) cosine similarity is calculated; (7) rubric-oriented and rule-based evaluation logic is applied; and (8) the result is presented for teacher review.

Teacher question + reference answer + rubric → Student file submission → Text extraction → Preprocessing → Sentence-BERT embeddings → Cosine similarity → Rubric-oriented evaluation → Result and teacher review

Fig. 1. Workflow of the proposed AI-assisted academic assignment evaluation system.

VI. IMPLEMENTATION AND FUNCTIONAL TESTING

The project was implemented as a web application with a React/Vite frontend and a Python FastAPI backend. SQLite was selected for storing application records. The evaluation component uses a pretrained Sentence-BERT model to encode text and cosine similarity to compare the semantic representations of student and reference answers. The model was not fine-tuned for the project.

Functional testing was carried out to verify the application workflow. The testing focus was whether the main application operations could be performed, including entering assessment information, submitting assignment content, processing text for evaluation, and displaying the resulting output. The available project information confirms functionality testing; it does not provide a labelled test dataset, human-marked benchmark, sample size, measured response time, accuracy, precision, recall, F1-score or agreement statistic. Such numerical results are therefore not reported.

1. *Functional Test Summary*

Test area	Purpose of verification	Evidence status
Assessment input	Verify entry of question, reference answer and rubric.	Included in functional testing scope.
Assignment submission	Verify that student assignment content can be submitted.	Included in functional testing scope.
Text processing	Verify extraction/preparation of submitted text for evaluation.	Included in functional testing scope.
Semantic comparison	Verify use of Sentence-BERT embeddings and cosine similarity.	Implemented evaluation methods.
Result presentation	Verify that the application displays the evaluation of output.	Included in functional testing scope.
Human review	Allow teacher reviews of generated evaluation.	Teacher oversight is part of the intended workflow.

2. *Discussion*

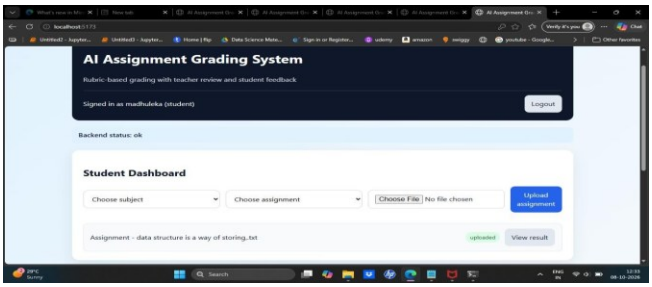
The functional implementation demonstrates how semantic text comparison can be integrated into an assignment-evaluation application. The use of Sentence-BERT allows the system to compare meaning beyond exact word overlap, while the rubric-oriented workflow retains the teacher's criteria as the basis for academic assessment. The separation of frontend, backend, model processing, and database responsibilities also provides a modular structure for further development.The present results should be interpreted within the limits of the testing performed. Functional verification demonstrates that application operations can be carried out; it does not establish that the generated marks are equivalent to expert human grading. A future experimental evaluation should use a sufficiently large collection of student answers independently graded by multiple teachers. Suitable measures may include mean absolute error, Pearson or Spearman correlation, quadratic weighted kappa and agreement by rubric criterion, depending on the scoring scale and study design.

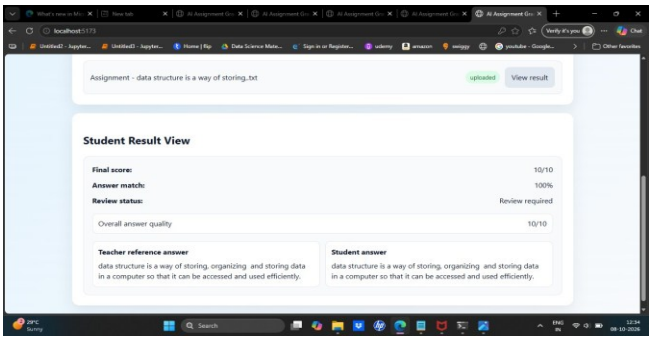
VII. LIMITATIONS AND FUTURE SCOPE

The current system uses a pretrained Sentence-BERT model without project-specific fine-tuning. Consequently, its semantic representations may not fully capture specialized terminology, subject-specific reasoning, or the expected depth of an answer in every discipline. Cosine similarity measures semantic closeness and cannot independently verify factual correctness. A response may be semantically similar while containing an incorrect statement, or it may be correct while using a different explanation from the reference answer.

The project has been functionally tested, but a quantitative comparison with teacher-assigned marks has not yet been documented. Further work should include a curated dataset of assignments and teacher-validated scores, inter-rater agreement analysis, threshold calibration and testing across different subjects, and answer lengths. Additional improvements may include clearer criterion-level explanations, support for more document formats, robust handling of scanned documents where OCR is required, security and privacy controls for student submissions, and a teacher correction interface. Any extension should be evaluated before being described as an implemented capability.

VIII. RESULT





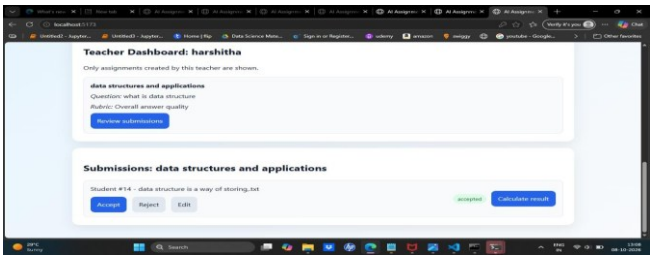


fig 2 Student Dashboard

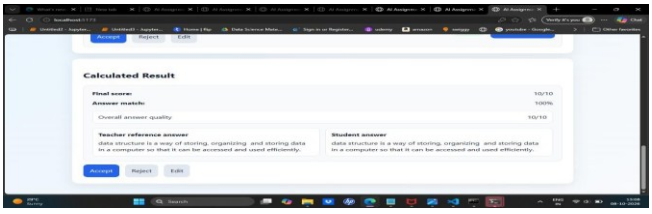


fig 3 Teacher Dashboard

IX. CONCLUSION

This paper presented an AI-assisted system for academic assignment evaluation using natural language processing and semantic similarity. The system combines teacher-provided questions, reference answers, and rubrics with student assignment submissions. Submitted text is processed by a Python FastAPI backend, represented using a pretrained Sentence-BERT model and compared with reference answers through cosine similarity. A rubric-oriented and rule-based workflow supports the evaluation, while a React/Vite interface and SQLite database provide the application structure.

The project demonstrates a functional integration of document-based submission, semantic response comparison, and teacher-facing evaluation support. Sentence-BERT is used without task-specific fine-tuning. Since the completed testing focused on functionality, this paper does not claim a measured grading accuracy or equivalence to human assessment. The system is best regarded as a decision-support application in which teachers retain final authority over marks. Future work will focus on quantitative validation with teacher-graded datasets, improving rubric-level explainability, and assessing reliability across subjects.

X. ACKNOWLEDGEMENT

The authors thank GSSS Institute of Engineering and Technology for Women, Mysuru, for providing the academic environment and support required to carry out this project. The authors also express their gratitude to the project guide and faculty members for their guidance and feedback throughout the development of the system.

REFERENCES

1. E. B. Page, “The imminence of…grading essays by computer,” Phi Delta Kappan, vol. 47, no. 5, pp. 238–243, 1966.
2. T. K. Landauer, P. W. Foltz, and D. Laham, “An introduction to latent semantic analysis,” Discourse Processes, vol. 25, no. 2–3, pp. 259–284, 1998.
3. Y. Attali and J. Burstein, “Automated essay scoring with e-rater® V.2,” The Journal of Technology, Learning, and Assessment, vol. 4, no. 3, 2006.
4. P. Phandi, K. M. A. Chai, and H. T. Ng, “Flexible domain adaptation for automated essay scoring using correlated linear regression,” in Proc. EMNLP, 2015, pp. 431–439, doi: 10.18653/v1/D15-1049.

5. K. Taghipour and H. T. Ng, "A neural approach to automated essay scoring," in Proc. EMNLP, 2016, pp. 1882–1891, doi: 10.18653/v1/D16-1193.
6. D. Alikaniotis, H. Yannakoudakis, and M. Rei, "Automatic text scoring using neural networks," in Proc. ACL, 2016, pp. 715–725, doi: 10.18653/v1/P16-1068.
7. M. A. Hussein, H. A. Hassan, and M. N. Nassef, "Automated language essay scoring systems: A literature review," PeerJ Computer Science, vol. 5, e208, 2019, doi: 10.7717/peerj-cs.208.
8. M. Mohler and R. Mihalcea, "Text-to-text semantic similarity for automatic short answer grading," in Proc. EACL, 2009, pp. 567–575.
9. 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, no. 1, pp. 60–117, 2015, doi: 10.1007/s40593-014-0026-8.
10. F. F. Lubis et al., "Automated short-answer grading using semantic similarity based on word embedding," International Journal of Technology, vol. 12, no. 3, pp. 571–581, 2021, doi: 10.14716/ijtech.v12i3.4651.
11. C. Ormerod et al., "Automated short answer scoring using an ensemble of neural networks and latent semantic analysis classifiers," International Journal of Artificial Intelligence in Education, vol. 33, pp. 467–496, 2023, doi: 10.1007/s40593-022-00294-2.
12. A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems, vol. 30, 2017.
13. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
14. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in Proc. EMNLP-IJCNLP, 2019, pp. 3982–3992, doi: 10.18653/v1/D19-1410.
15. H. Misgna, B.-W. On, I. Lee, and G. S. Choi, "A survey on deep learning-based automated essay scoring and feedback generation," Artificial Intelligence Review, vol. 58, art. no. 36, 2025, doi: 10.1007/s10462-024-11017-5.
16. "A survey of automated essay scoring: Challenges, advances, and future," Neurocomputing, vol. 650, art. no. 130916, 2025, doi: 10.1016/j.neucom.2025.130916.
17. P. W. Foltz, D. Laham, and T. K. Landauer, "The intelligent essay assessor: Applications to educational technology," Interactive Multimedia Electronic Journal of Computer-Enhanced Learning, vol. 1, no. 2, 1999.
18. S. Dikli, "An overview of automated scoring of essays," The Journal of Technology, Learning, and Assessment, vol. 5, no. 1, 2006.
19. C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge, U.K.: Cambridge University Press, 2008.
20. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in Proc. ICLR Workshop, 2013.
21. J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in Proc. EMNLP, 2014, pp. 1532–1543.
22. M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in Proc. ICML, 2015, pp. 957–966.