iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: IV Month of publication: April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69617

Automatic Subjective Answer Evaluation

Dr. Dipak D. Bage[1], Tina H. Deore[2], Samarth S. Abak[3], Shruti D. Godse[4], Vishakha P. Mandawade[5], Dr. Aniruddha S. Rumale[6]

[1]Associate Professor, [2, 3, 4, 5]Research Scholar, [6]Professor, Department of Information Technology
Sandip Foundation's Sandip Institute of Technology and Research Centre, Nashik

Abstract: Automatic evaluation of subjective answers has become a vital area of research due to its potential to reduce the manual effort required in educational assessments. This paper presents an advanced system for the automatic evaluation of handwritten subjective answers, integrating Optical Character Recognition (OCR), Natural Language Processing (NLP), and semantic similarity techniques. The system employs the Google Cloud Vision API to extract textual data from handwritten answer sheets with high accuracy. Extracted responses undergo preprocessing, including spell correction, and are semantically compared with ideal answers using both BERT and fine-tuned SBERT models. To enhance grading reliability, a custom contrastive learning mechanism is implemented for SBERT fine-tuning, using student-ideal answer pairs. The evaluation is performed via a Flask-based backend, which also supports training workflows through API endpoints. Feedback and marks are generated based on semantic similarity and model confidence. This system demonstrates an effective solution for automating subjective answer assessment with a high degree of flexibility and accuracy, particularly for handwritten inputs. The system supports both real-time evaluation and model customization, offering educators the flexibility to retrain models using domain-specific datasets. A user-friendly web interface allows for seamless uploading of answer images, configuration of model settings, and visualization of results. Additionally, the system integrates a secure user authentication module for access control, enabling personalized model training and usage history tracking. Experimental results demonstrate that the fine-tuned SBERT model significantly improves semantic alignment with ground-truth answers, especially in the context of varied handwriting styles and non-standard grammar.

Keywords: Subjective Answer Evaluation, Handwriting Recognition, Google Cloud Vision API, SBERT, Optical Character Recognition (OCR), Natural Language Processing (NLP), Flask API.

I. INTRODUCTION

In educational settings, manually evaluating subjective answers can lead to bias, inconsistencies, and delays. The rise of deep learning and natural language processing (NLP) presents an opportunity to automate this process, ensuring more objective, accurate, and efficient grading. This research proposes a deep learning-based system that utilizes Optical Character Recognition (OCR) and semantic similarity models to automatically evaluate handwritten subjective responses. The system follows a two-step approach. First, handwriting recognition is performed using OCR, specifically the Google Cloud Vision API, to extract text from handwritten answer sheets and convert it into machinereadable format. To enhance accuracy, automated spell correction is applied to rectify any errors in the extracted text. Second, the processed student response is compared with a model answer using advanced NLP techniques. The Sentence-BERT (S-BERT) model is employed to evaluate the semantic similarity between responses, ensuring that meaning and contextual relevance are properly assessed. Additionally, fuzzy matching and token-level comparisons further refine the grading process by capturing lexical similarities. This approach ensures a human-like evaluation by assessing the context, meaning, and coherence of responses rather than merely matching keywords. Traditional assessment systems rely heavily on human evaluators, making them prone to subjectivity and variability. In contrast, this automated system enhances fairness and efficiency by leveraging deep learning models trained for sentence similarity, token-based analysis, and contextual understanding. Furthermore, OCR technology has significantly evolved in recent years. Earlier OCR methods were confined to high-performance desktop environments, requiring substantial processing power and memory. However, modern cloud-based OCR solutions, such as Google Cloud Vision API, provide scalable and highly accurate text extraction. By integrating deep learning-driven OCR and NLP methodologies, this research presents a robust, automated grading system that minimizes human intervention while maintaining consistency and accuracy.

II. LITERATURE SURVEY

In [1], the development of natural language processing (NLP) and optical character recognition (OCR) methodologies for the automated evaluation of subjective responses. This article evaluates several natural language processing methodologies on prominent datasets, including the SICK dataset, STS benchmark, and Microsoft Paraphrase Identification. They may assess optical character recognition methodologies using MNIST, EMNIST, IAM datasets, and others.

According to [2], the examination of the research uncovers diverse methodologies for assessing subjective response sheets. The system's benefit is that it uses a weighted average of the most precise approaches to get the optimum outcome. TESA is a methodical and dependable technique that facilitates assessors' responsibilities and delivers faster and more effective results. This technology generates a dependable, resilient, and evident rapid reaction time.

According to [3], a voice-over-guided system to teach visually impaired individuals how to compose multilingual letters. The technology constantly observes and records the learner's strokes, while a voice-over guide provides appropriate suggestions. It will also notify if the student executes an incorrect stroke or positions the stylus outside the permissible range. This method may effectively teach any alphabet and language, allowing visually challenged students to engage in writing. They have created a language-agnostic algorithm to assist visually challenged individuals in writing multilingual alphabets. In this paper, they have implemented a voice-over guiding system in the educational process, which removes the necessity for heavy or costly equipment installations. The system integrates machine learning algorithms to assess the progress of learners. They evaluate an effective and user-focused system through usability testing.

In [4], an advanced deep learning architecture that combines convolutional neural networks (CNN) and bidirectional long short-term memory (BiLSTM) to accurately find and grade handwritten responses, just like an expert grader would. The model is specifically designed to evaluate responses consisting of 40 words, 13 of which are lengthy. They constructed the model using several methodologies, which involved modifying parameters, deep layers, neuron count, activation functions, and bidirectional LSTM layers. They systematically adjusted each parameter several times and included or eliminated layers, LSTMs, or nodes to identify the most efficient and best model.

In [5], the system utilizes a personal computer, a portable scanner, and application software to automatically correct handwritten response sheets. The Convolutional Neural Network (CNN), a machine learning classifier, processes scanned pictures for handwritten character identification. They developed and trained two CNN models using 250 photos from students at Prince Mohammad Bin Fahd University. The suggested approach would ultimately provide the student's final score by juxtaposing each categorized response with the right answer.

According to [6], the first model employs deep convolutional neural networks (CNNs) for feature extraction and a fully connected multilayer perceptron (MLP) for word categorization. The second model, termed SimpleHTR, employs convolutional neural network (CNN) and recurren t neural network (RNN) layers to extract data from images. They also offered the Bluechet and Puchserver models for data comparison. Owing to the scarcity of accessible open datasets in Russian and Kazakh languages, they undertook the task of compiling data that included handwritten names of nations and towns derived from 42 distinct Cyrillic words, inscribed over 500 times in various handwriting styles.

In [7], a self-supervised, feature-based categorization problem that is capable of autonomously fine- tuning for each inquiry without explicit supervision. The use of information retrieval and extraction (IRE) and natural language processing (NLP) techniques, together with semantic analysis for self- evaluation in handwritten text, creates a set of useful character traits. They evaluated their methodology on three datasets derived from diverse fields, with assistance from students of varying age groups.

In [8], they discuss the needs, relevant research towards handwritten recognition, and how to process it. They outline the steps and stages used in the recognition of Kannada handwritten words. The main aim of proposed work is to identify Kannada handwritten answer written in answer booklets and to solve recognition problem by using machine learning algorithms. System provides a detailed concept on pre-processing, segmentation, and the classifier used to develop systematic OCR tool.

Kumar, Munish, et al. [9], discuss the necessary conditions, relevant studies on handwriting identification, and techniques for processing. They outline the procedures and phases involved in identifying Kannada handwritten words. The primary goal of the proposed study is to recognize Kannada handwritten responses in answer booklets and address the identification challenge using machine learning methods. The system provides a comprehensive framework for pre-processing, segmentation, and classification that is used in the development of a systematic OCR tool.

Mukhopadhyay, Anirban, et al. [10], Information given by one form-based and two texture-based data characteristics are combined from handwritten text images using classifier mixture techniques for script recognition (word-level) purposes. Based on the confidence scores supplied by the Multi- Layer Perceptron (MLP) classifier, the word samples from the specified database are listed. For this pattern recognition problem, major classifier combination techniques such as majority voting, Borda count, sum rule, product rule, max rule, Dempster-Shafer (DS) combination rule and secondary classifiers are evaluated.

Summary Table

Author(s)	Title	Methodology	Algorithms	Limitations
Souibgui, Mohamed Ali, et al.[11]	Docentr: An End-to-End Document Image Enhancement Transformer	Proposes an end-to-end Transformer model specifically designed for enhancing document images; includes techniques for image denoising, enhancement, and text clarity improvement	Transformer, U-Net Architecture, Image Enhancement Techniques	May require extensive computational resources and training data; performance may vary based on the quality and type of input images
Shailesh Acharya Ash ok Kumar Pant Prashnna Kumar Gyawali [12]	Deep Learning Based Large Scale Handwritten Devanagari Character Recognition	Dataset increment, Dropout layers, Stochastic gradient descent (SGD) with momentum, Local response normalization, ReLU activation.	Model A: Deep Convolutional Neural Network (CNN), Model B: Shallow Convolutional Neural Network (CNN),	High visual similarity between some characters, leading to ambiguity, Variability in handwritten styles across individuals.
			Overlapping kernel scheme, non-overlapping kernel scheme.	
Ali, Amani Ali Ahmed et al. [13]	Intelligent Handwritten Recognition Using Hybrid CNN Architectures Based-SVM Classifier with Dropout	Utilizes a hybrid CNN architecture combined with an SVM classifier for handwritten recognition; includes dropout techniques for regularization and model robustness	CNN (Convolutional Neural Network), SVM (Support Vector Machine), Dropout Regularization	May face Challenges with varying handwriting styles and quality; requires careful tuning of dropout rates and model parameters
Teslya, Nikolay et al. [14]	Deep Learning for Handwriting Text Recognition: Existing Approaches and Challenges	A comprehensive review of deep learning techniques for handwriting text recognition; includes analysis of various models, architectures, and their performance	CNN (Convolutional Neural Network), RNN (Recurrent Neural Network), LSTM (Long Short-Term Memory), Transformer	Challenges with diverse handwriting styles, data variability, and model generalization; requires extensive datasets and computational resources

Alrobah, Naseem et al. [15]	A Hybrid Deep Model for Recognizing Arabic Handwritten Characters	Development of a hybrid deep learning model combining CNNs and RNNs to recognize Arabic handwritten characters; includes preprocessing, feature extraction, and classification	CNN (Convolutional Neural Network), RNN (Recurrent Neural Network), LSTM (Long Short-Term Memory)	Performance may be affected by variations in handwriting styles and character shapes; requires extensive training data

III. PROPOSED SYSTEM

The "Automatic Subjective Answer Evaluation System" is designed with a clear and organized process to fairly assess student responses, whether they're handwritten or typed. It starts by taking in the student’s answer—either as an image or text—along with the correct (model) answer and the total marks available. If the input is an image, the system uses Google Cloud Vision API to extract the text and correct any spelling mistakes. Once the text is ready, it goes through a series of preparation steps like converting everything to lowercase, removing punctuation, breaking the text into tokens, and normalizing the words.

For the actual evaluation, the system uses a fine-tuned Sentence-BERT model to understand the meaning of the student’s answer and compare it to the model answer. It also incorporates fuzzy matching and token matching to measure how closely the responses align. All these methods are blended to generate a similarity score. Based on that score, the system calculates how many marks the student earns and offers constructive feedback on their answer.

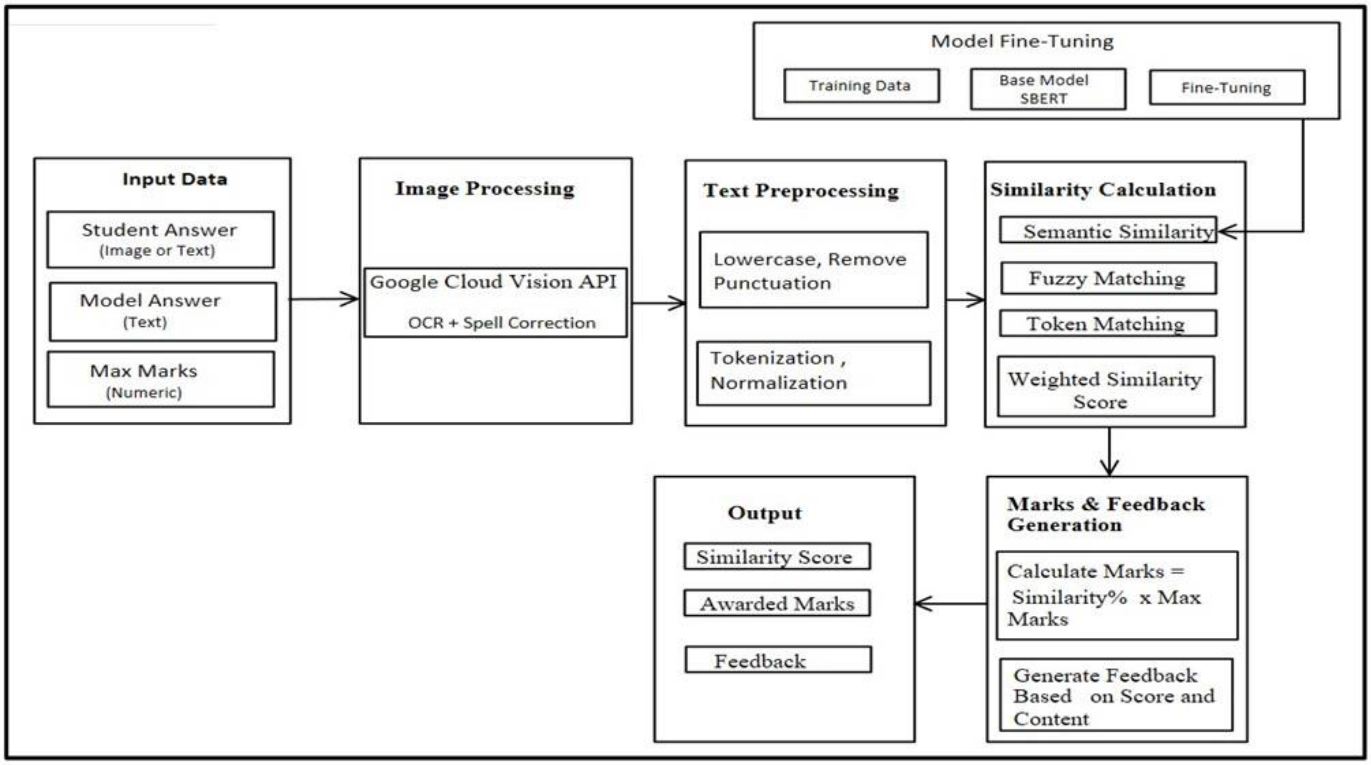


Fig.1:Proposed System Architecture

A. System Workflow:

The system operates through a well-organized workflow designed to evaluate student responses accurately. It begins with the Input Data Collection stage, where users either upload an image of a handwritten answer or directly enter text. Along with this, a reference answer and the maximum possible marks for the question are also submitted.

If the student’s response is handwritten, the system initiates Image Processing. Here, the Google Cloud Vision API is employed for Optical Character Recognition (OCR) to convert the handwritten text into a digital format. To correct any inaccuracies that may result from OCR, spell-checking algorithms are applied.

Once the text has been converted to digital format, the system proceeds to the Text Preprocessing stage. This stage prepares the data for analysis by converting all characters to lowercase, stripping away punctuation, and applying techniques like tokenization and normalization to ensure consistency across all inputs.

At the heart of the system is the Similarity Calculation process. This phase evaluates how closely the student’s answer aligns with the reference answer. Semantic similarity, evaluated through SBERT embeddings to understand sentence meaning, makes up 50% of the overall score. Fuzzy matching techniques, accounting for 30%, identify character-level matches, while token-level similarity, weighted at 20%, checks for overlapping key terms. These factors are combined into a single weighted similarity score.

The next step, Marks and Feedback Generation, uses the similarity percentage to compute the final score based on the maximum possible marks. It also provides tailored feedback aimed at helping students understand how they can improve. Finally, the Output Display shows the similarity percentage, the awarded marks, and detailed feedback. This gives students a clear and structured overview of their performance and areas needing attention.

B. *Model Explanation:*

SBERT (Sentence-BERT): We use the SBERT model to embed and compare sentences. The cosine similarity is calculated to identify the closest match. The model used in this project is paraphrase-mpnetbase-v2. This model is a transformer-based architecture designed for capturing sentence-level meaning and understanding the context. It is particularly useful for comparing academic-style answers because it evaluates the semantic relationship rather than exact word matching. The model generates highdimensional embeddings for both the model and the student's answer. By computing cosine similarity between these embeddings, the system can identify how semantically similar the answers are.

C. *Matching Process:*

The evaluation approach incorporates three key components to assess textual similarity, each contributing a weighted score. Semantic Similarity, accounting for 50% of the overall weight, utilizes sentence-level comparison through SBERT embeddings. This method captures the conceptual meaning of the text, allowing it to handle paraphrasing and diverse sentence structures effectively. Fuzzy Matching contributes 30% of the score by applying character-level similarity techniques, such as sequence matching algorithms. This enhances tolerance for minor spelling variations and OCR-related errors, making the evaluation more robust, especially when dealing with inputs from handwritten or scanned sources. Finally, Token Matching, with a 20% weight, focuses on word-level comparison to ensure that essential terminology is present. This component helps identify key concept terms regardless of how the sentence is constructed, providing an additional layer of evaluation beyond the semantic understanding.

D. *User Interface (GUI):*

The system features a comprehensive and intuitive web-based user interface developed using HTML, CSS, and JavaScript, with Flask serving as the backend framework. It is organized into multiple functional tabs to ensure seamless navigation and interaction. The "Answer Upload" tab enables users to submit handwritten answer sheets in image formats such as JPG or PNG. Upon submission, the system utilizes the Google Cloud Vision API to extract textual content through Optical Character Recognition (OCR). Once evaluation is complete, the "Result Display" tab presents a structured view of the semantic similarity scores, predicted marks, and feedback for each answer. To support custom use cases, the "Model Training" tab provides an interface for fine-tuning the SBERT model using user-provided JSON datasets, invoking the training process through Flask API endpoints. Additionally, the system incorporates a secure user authentication mechanism, enabling user registration and login with hashed password storage. This allows for personalized sessions, ensuring that training history and evaluation data remain user-specific. The overall interface is designed to be accessible and efficient, empowering both educators and researchers to perform automated subjective answer evaluation with minimal technical effort.

E. *Algorithm***:**

Input:

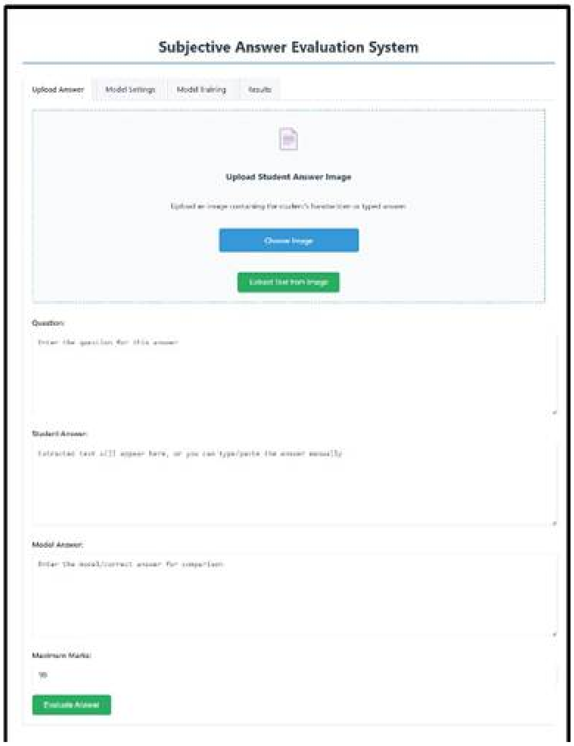
image_path: Path to the image containing the student's handwritten answer. model_text: The reference (model) answer for comparison.

```
Output: Final Score: Similarity score displayed on the GUI.
Step 1: Read each test instance from (Ts_Instnace from Ts)
Step 2:     TsIns $= \sum_{k=0}^{n}\{Ak\ldots An\}$
Step 3: Read each train instance from (Tr_Instnace from Tr)
Step 4:     $TrIns = \sum^{n}\{Aj\ldots\ldots Am\}$
$j$=0
Step 5:  $w$ = WeightCalc (TsIns, TrIns)
Step 6:  if (w >= T)
Step 7: Forward feed layer to input layer for feedback FeedLayer[]  {Tsf,w}
Step 8: optimized feed layer weight, Cweigt  FeedLayer [0]
Step 9:  Return Cweight
```

IV. RESULTS

A. *Final Outputs:*

As shown in Fig.1, the Subjective Answer Evaluation System is a digital platform designed to automate the evaluation of handwritten or typed student responses. It utilizes Optical Character Recognition (OCR) to extract text from uploaded images and allows users to input the corresponding question, model answer, and maximum marks. The system then facilitates the comparison between the student's response and the model answer, supporting consistent and efficient grading. This approach reduces manual workload and promotes objectivity in the assessment of subjective answers.



Upload an image containing the student's handwritten or typed answer

Question:
Enter the question for this answer

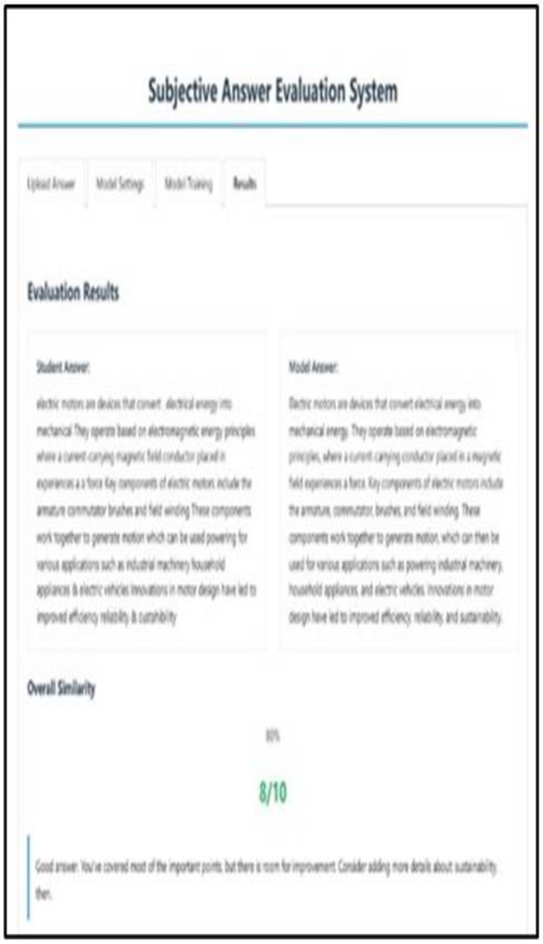
Student Answer:
Extracted text will appear here, or you can type/paste the answer manually

Model Answer:
Enter the model/correct answer for comparison

Maximum Marks:
10

Fig.1: Evaluation

As shown in Fig.2, a user interface was developed to demonstrate the Subjective Answer Evaluation System, allowing users to upload student answers, fine-tune semantic models, and view evaluation results. The system uses OCR to extract text from handwritten answers and compares them to a model answer using a fine-tuned Sentence-BERT model. Results include the student answer, model answer, semantic similarity score, and autogenerated feedback. For example, a response with 80% similarity earned 8 out of 10 marks, with feedback noting the coverage of key points. This showcases the system's ability to provide both quantitative and qualitative assessments for automating subjective answer evaluation.



Subjective Answer Evaluation System

Evaluation Results

Student Answer:

electric motors are devices that convert electrical energy into mechanical They operate based on electromagnetic energy principles where a current carrying magnetic field conductor placed in experiences a a force Key components of electric motors include the armature commutator brushes and field winding These components work together to generate motion which can be used powering for various applications such as industrial machinery household appliances & electric vehicles innovations in motor design have led to improved efficiency reliability & sustainability

Model Answer:

Electric motors are devices that convert electrical energy into mechanical energy. They operate based on electromagnetic principles, where a current carrying conductor placed in a magnetic field experiences a force. Key components of electric motors include the armature, commutator, brushes, and field winding. These components work together to generate motion, which can then be used for various applications such as powering industrial machinery, household appliances, and electric vehicles. Innovations in motor design have led to improved efficiency, reliability, and sustainability.

Overall Similarity

80%

8/10

Good answer. You've covered most of the important points, but there is room for improvement. Consider adding more details about sustainability then.

Fig.2: Results

B. Visual Representation:

These visualizations provide a comprehensive overview of automatic subject answer evaluation system.

1) *System Performance by Answer Category:* The comparative analysis reveals varying system accuracy across answer types, with highest performance for factual content (92%) and progressively lower but satisfactory results for more complex responses (conceptual: 85%, analytical: 76%, essay: 70%, creative: 64%)

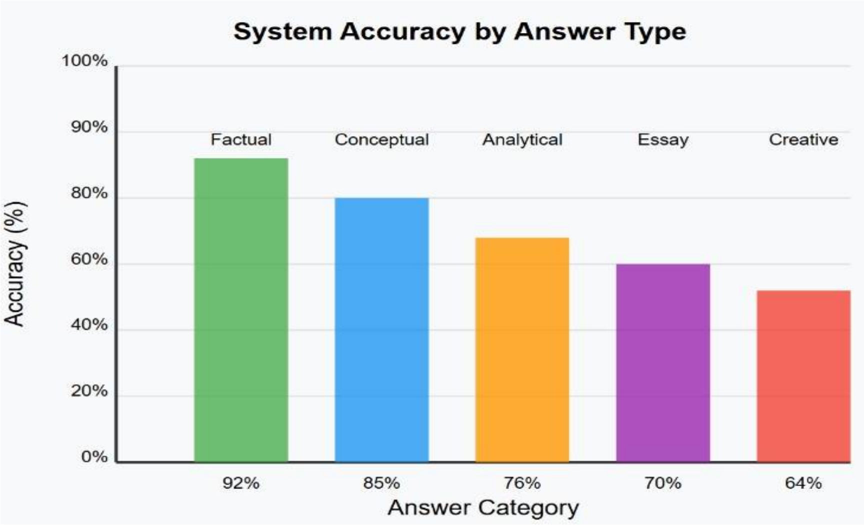


Fig. 3: System performance by answer category

2) Contribution of Different Similarity Metrics:

This bar chart illustrates the weighting factors used in your final scoring algorithm, showing that semantic similarity (50%) contributes most heavily to the final score, followed by fuzzy matching (30%) and token-level matching (20%).

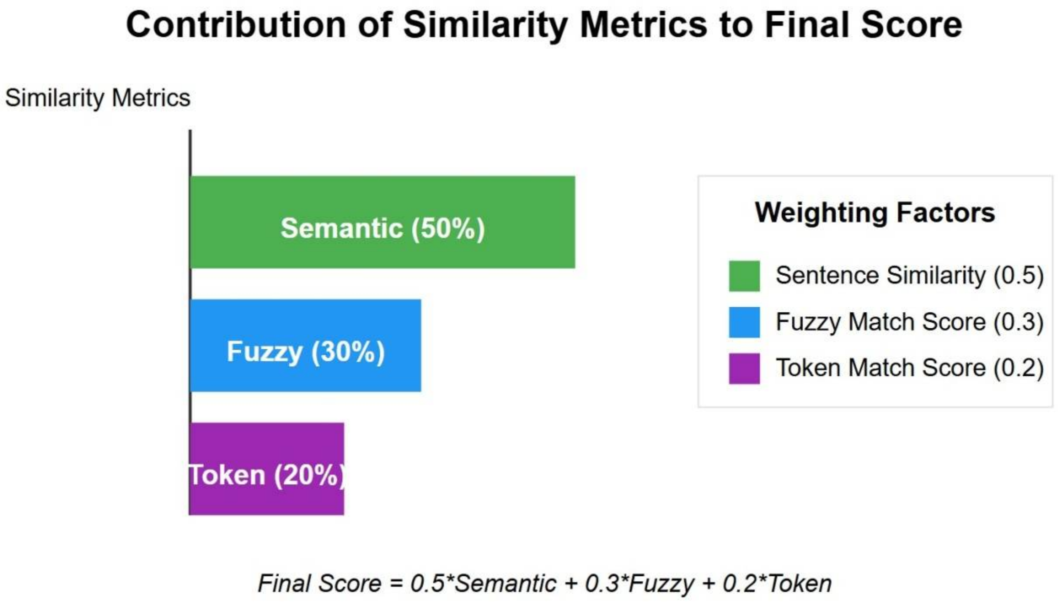


Fig. 3: Contribution of different similarity metrics

3) Feedback Thresholds Visualization:

This gradient chart visualizes how different similarity score ranges correspond to specific feedback messages, showing the threshold values at 50%, 75%, and 90% that determine which feedback statement is provided to students.

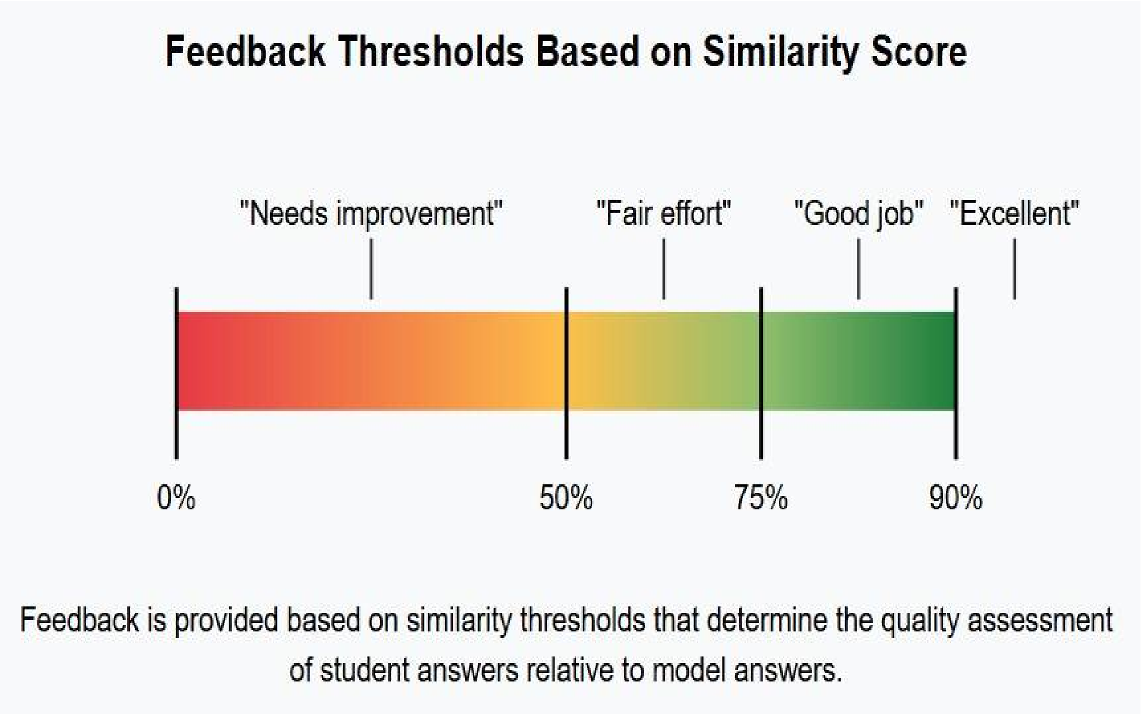


Fig. 5: Feedback thresholds visualization

4) Correlation Between Similarity Score and Grading Accuracy:

The scatter plot demonstrates a strong positive correlation ($R^2 = 0.92$) between algorithmcalculated similarity scores and actual grading accuracy, validating the system's effectiveness for automated assessment.

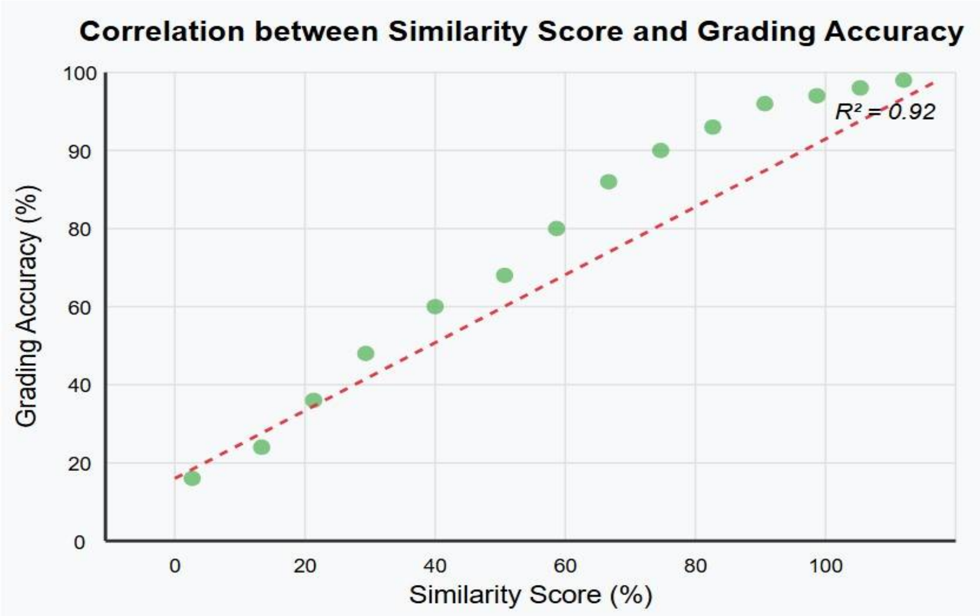


Fig. 6: Correlation between similarity score and grading accuracy

V. CONCLUSION

Our proposed system significantly improves upon the base paper by addressing key limitations in handwritten subjective answer evaluation. First, it goes beyond basic character recognition by incorporating semantic understanding to evaluate the meaning and context of answers. Second, it includes spelling correction to enhance the accuracy of text processing. Third, it mitigates OCR related errors through the use of multiple text matching techniques.

Additionally, the system eliminates manual grading bias by automating the evaluation process, ensuring fairness and consistency. Finally, it enhances user accessibility with a user-friendly interface that allows seamless answer uploads and result visualization. These advancements collectively make the system accurate and effective.

REFERENCES

[1] Desai, Madhavi B., et al. "A Survey on Automatic Subjective Answer Evaluation." Advances and Applications in Mathematical Sciences 20.11 (2021): 2749-2765.

[2] Singh, Shreya, et al. "Tool for evaluating subjective answers using AI (TESA)." 2021 International Conference on Communication information and Computing Technology (ICCICT). IEEE, 2021.

[3] Islam, Muhammad Nazrul, et al. "A multilingual handwriting learning system for visually impaired people." IEEE Access (2024).

[4] Rahaman, Md Afzalur, and Hasan Mahmud. "Automated evaluation of handwritten answer script using deep learning approach." Transactions on Machine Learning and Artificial Intelligence 10.4 (2022).

[5] Shaikh, Eman, et al. "Automated grading for handwritten answer sheets using convolutional neural networks." 2019 2nd International conference on new trends in computing sciences (ICTCS). IEEE, 2019.

[6] Nurseitov, Daniyar, et al. "Classification of handwritten names of cities and handwritten text recognition using various deep learning models." arXiv preprint arXiv:2102.04816 (2021).

[7] Rowtula, Vijay, Subba Reddy Oota, and C. V. Jawahar. "Towards automated evaluation of handwritten assessments." 2019 International Conference on Document Analysis and Recognition (ICDAR). IEEE, 2019.

[8] Sampathkumar, S. "Kannada Handwritten Answer Recognition using Machine Learning Approach." (2022).

[9] Kumar, Munish, et al. "Character and numeral recognition for non-Indic and Indic scripts: a survey." Artificial Intelligence Review 52.4 (2019): 2235-2261.

[10] Mukhopadhyay, Anirban, et al. "A study of different classifier combination approaches for handwritten Indic Script Recognition." Journal of Imaging 4.2 (2018): 39.

[11] Souibgui, Mohamed Ali, et al. "Docentr: An end-to-(ICPR). IEEE, 2022.

[12] Acharya, Shailesh, et al. "Deep Learning Based Large Scale Handwritten Devanagari Character Recognition."

[13] Ali, Amani Ali Ahmed, and Suresha Mallaiah. "Intelligent handwritten recognition using hybrid CNN Journal of King Saud University-Computer and Information Sciences 34.6 (2022): 32943300.

[14] Teslya, Nikolay, and Samah Mohammed. "Deep Open Innovations Association (FRUCT). IEEE, 2022.

[15] Alrobah, Naseem, and Saleh Albahli. “A Hybrid Deep Model for Recognizing Arabic Handwritten Characters”.