An Automated Approach for Evaluating Answer Scripts using Machine Learning and Natural Language Processing

Dr. Venkata Satya Santhi Somisetti[1], Y Satya Vijaya Laxmi[2], K Sreeja[3], Naga Heera Agarwal[4], A Dinesh[5], A Samuel[6]

[1]Associate Professor of CSE (AIML & DS) Dept,
[23456]Student of CSD Computer Science and Engineering (Data Science), Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, India

---

Abstract - *Traditional manual grading of student answer scripts is time-consuming, error-prone, and inconsistent, especially for descriptive responses. This paper presents an automated evaluation system using machine learning (ML) and natural language processing (NLP) techniques. The process begins with uploading a key script, followed by entering marks for reference. The total number of students is recorded, along with absentees and detained students. Student scripts are then uploaded sequentially for evaluation. The system extracts text from the scripts, preprocesses the data, and applies the T5 model along with Sentence-BERT (SBERT) to compute similarity scores. Based on this, a percentage score is calculated, and a grade is assigned according to predefined criteria. Finally, an Excel report is generated, listing each student's roll number and grade. The proposed system achieves an accuracy of 93.06 percent in similarity-based evaluation, minimizing manual effort, improving grading consistency, and enhancing accuracy in automated evaluation for educational assessments.*

Key Words: Answer scripts evaluation, Natural language processing, Machine learning, LLM Whisperer, T5 model, Sentence Embedding, Similarity score

1. INTRODUCTION

The integration of Machine Learning (ML) and Natural Language Processing (NLP) has brought significant improvements to many fields, including education. One key area benefiting from automation is the grading of student answer scripts, which is traditionally time-consuming and inconsistent, especially for long, descriptive responses. Manual grading can lead to errors, making it difficult to ensure fairness. With advancements in technology, automated grading systems simplify this process by reducing human effort, improving accuracy, and ensuring unbiased evaluation. By using ML and NLP techniques, these systems analyze, compare, and grade answers efficiently, making the assessment process faster and more effective.

This paper presents an Automated Answer Script Evaluation System that reduces manual effort and improves grading accuracy. The process begins with uploading a key answer script and entering marks for reference. The total number of students is recorded, including absentees and detained students. Then, student answer scripts are uploaded one by one for evaluation. The system extracts text from each script, processes the data, and applies T5 summarization to shorten long answers while preserving key information. After summarization, the T5 question-answering approach is used to refine the extracted content. The system then compares the student' s response with the key script using SBERT with cosine similarity to measure accuracy. Based on the similarity score, marks are calculated, and grades are assigned. Finally, the results are stored in an Excel report, listing each student's roll number and grade.

This research focuses on:

- Developing an automated grading system using ML and NLP.
- Extracting, summarizing (T5), and refining answers using the T5 question-answering approach.
- Using Sentence-BERT (SBERT) and cosine similarity to compare student answers with the key script.
- Generating an Excel report to store student grades efficiently.

Unlike traditional grading methods, which can be slow and inconsistent, this system ensures quick, fair, and accurate evaluation while reducing manual work. It also helps manage a large number of answer scripts efficiently, improving the overall assessment process.

2. LITERATURE SURVEY

Multiple approaches have been explored to develop effective automated answer script evaluation systems. One research work explored automated answer evaluation using NLP and machine learning. One research study proposed automated answer evaluation using NLP techniques such as word embeddings and machine learning methods like cosine similarity, along with deep learning models including BERT and Convolutional Neural Network (CNN). This approach significantly enhances consistency by analysing semantic similarity between student answers and reference responses. It is particularly effective for evaluating short-answer questions. However, the system is limited in its ability to handle long or descriptive answers, which may lead to inconsistencies in evaluation for such responses[1]. The GradeAid-TF-IDF (Term Frequency–Inverse Document Frequency) model with ML regression improves short-answer grading by using semantic and lexical features. However, it struggles with long answers and may produce inconsistent results for lengthy responses[2].

One more approach utilizes NLP-based word embeddings to process and evaluate responses. This method can handle large volumes of textual data efficiently, improving accuracy and performance. Despite these advantages, it is primarily designed for short- answer grading, limited to 40 words, and may not perform well with longer, more complex responses[3]. One approach for automatic short- answer grading involves utilizing various preprocessing techniques such as tokenization, stop words removal, and lemmatization, which help improve the representation of answers. A simple measure of text similarity, can be employed to evaluate the similarity between ideal and student responses. More machine learning models may further enhance performance by addressing challenges like spelling errors[4].

One more approach is to automate the evaluation of subjective papers is by using machine learning and natural language processing (NLP) techniques such as Word2Vec, Word Mover's Distance (WMD), Cosine Similarity, and Multinomial Naive Bayes (MNB). This method evaluates descriptive answers by utilizing solution statements and keywords to predict grades. The results show an accuracy of 88%. However, this approach struggles to accurately evaluate answers that require deeper contextual understanding[5]. One approach for automatic short-answer grading involves utilizing various preprocessing techniques such as tokenization, stop words removal, and lemmatization, which help improve the representation of answers. Cosine similarity, a simple measure of text similarity, can be employed to evaluate the similarity between ideal and student responses. More advanced similarity measures or machine learning models may further enhance performance by addressing challenges like spelling errors. However, while cosine similarity and basic graph-based techniques (such as using WordNet graphs) work well for structured comparisons, they struggle to accommodate spelling errors and fail to assign partial credit for incomplete or partially correct answers[6]. One more approach involves an NLP-based evaluation system that extracts text from answer scripts and generates scores. While effective in many cases, its accuracy is limited to around 70-80%, as it may not fully grasp the context or implied meaning in student responses[7].

These research works highlight the challenges and advancements in automated answer evaluation, including difficulties in handling open-ended questions, processing handwritten responses, understanding complex contexts, and providing partial credit for partially correct answers. Our proposed system aims to overcome these challenges by integrating NLP-based techniques, enhanced text recognition for handwritten scripts, structured evaluation methods for open-ended responses, improved contextual understanding, and an adaptive scoring mechanism to ensure a more accurate and fair assessment process.

3. METHODOLOGY

The system is divided into four main phases: Text Extraction, Preprocessing, Score Generation, and Answer Script Grading. These phases work together to automate the evaluation of student answer scripts. Starting with extracting text from PDFs and ending with generating final grades, each phase plays a key role in ensuring accurate and efficient assessment. The final output is an Excel report that contains the grades of each student.

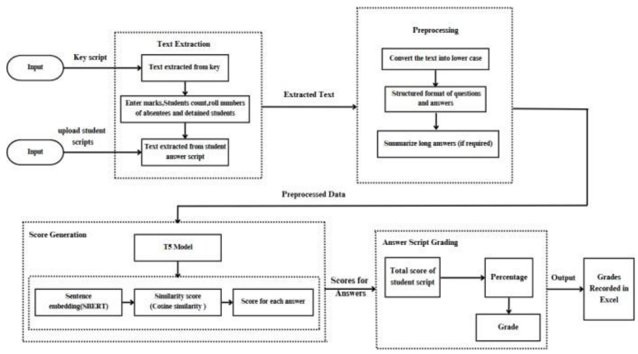

Fig -1: Automated answer script evaluation system

Phase 1: Text Extraction

In this phase, the key script and student scripts in PDF format are given as input. These files are sent to an external API called llmwhisperer-api to extract text from the PDFs.

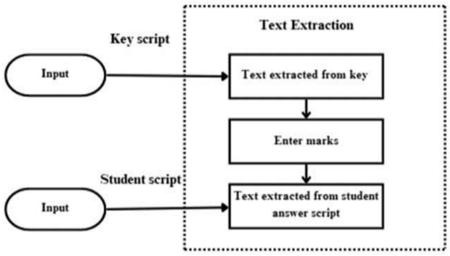

Fig -2: Text Extraction

Initially, the system extracts text from the answer key. Then, the user enters marks based on the extracted questions. After entering the marks, the user is required to enter the total number of students, followed by the roll numbers of absentees and detained students. Once this information is provided, the system asks the user to upload the PDF files of the student's script one by one. The system then extracts text from each student's answer script for comparison and evaluation. This process helps assess student responses efficiently by automating text extraction. The system sends a request to the API to retrieve the extracted text. If successful, the extracted text is returned; otherwise, an error message is displayed.

Phase 2: Preprocessing

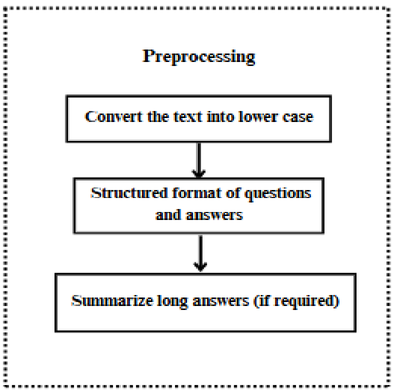

Fig -3: Preprocessing

In this phase, the text is properly processed to ensure clarity and consistency. First, all the text is converted to lowercase to avoid issues related to case sensitivity when comparing answers. Next, the system extracts and organizes the questions and answers into a structured format that includes the original question, the correct answer, and the student's response. If any answer is too long (more than 512 words), a technique called hierarchical summarization is used. This method breaks the long text into smaller parts and summarizes each part, making sure important details are not lost. This helps the model handle longer responses more effectively. After the text is preprocessed and summarized if needed, it is passed to the T5 model. Overall, this preprocessing phase ensures that the input data is clean, structured, and ready for accurate evaluation.

Phase 3: Score Generation

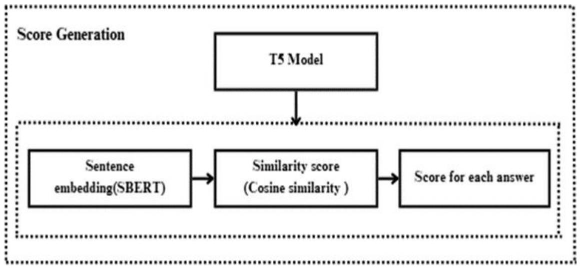

Fig -4: Score Generation

In this phase, the scoring mechanism is used to evaluate student answers based on their similarity to the correct answers using the T5 model and SBERT. First, the preprocessed and structured data containing questions, correct answers, and student responses is given as input to the T5 model for processing. After that, SBERT generates sentence embeddings to convert both the answer key and the student's response into numerical vectors that capture the meaning of the text. These vectors are compared using cosine similarity, which measures how close the two responses are in meaning. A higher similarity score means the student's answer is closer to the correct answer, and this score is used to assign marks. At this stage, the system generates scores for all the answers in the student's script. Overall, it provides a reliable way to automate answer evaluation and grading.

In this phase, we considered the answers generated by T5 model as Akey and Astudent , while Skey and Sanswer are the embeddings generated using SBERT.

$$\text{Cosine\_similarity SBERT} = \frac{S_{key} \cdot S_{student}}{\|S_{key}\| \cdot \|S_{student}\|} \quad (1)$$

Where:

S_{key} . $S_{student}$ represents the dot product of the two vectors.

Cosine similarity usually ranges from -1 to 1 but in this proposed system negative values are considered zero. A higher similarity value means the student's answer is very close to the actual answer while a lower value shows less similarity. The final score is based on this so a high score means the answers match closely and a low score means they do not.

Question	Actual Answer	Student answer	Obtained marks	Total score
Define AI?	Artificial (AI) Intelligence refers to the development of computer systems that can perform tasks that typically require human intelligence. These tasks include learning, problem-solving, reasoning, perception, and language understanding. AI systems use algorithms, data, and computing power to simulate intelligent behavior, enabling machines to think and learn like humans.	AI is a technology that makes machines think and learn like humans.	1	5
		Artificial Intelligence refers to the development of computer systems that can simulate human intelligence. AI systems use algorithms and data to perform tasks like learning and problem solving.	2.5	
		Artificial Intelligence is a field of computer science that focuses on creating intelligent machines. AI systems can perform tasks that typically require human intelligence, such as learning, problem-solving, and reasoning. These systems use algorithms, data, and computing power to simulate intelligent behavior. AI has transformed various industries and continues to revolutionize the way we live and work.	5	
		AI is a technology that enables machines to think and learn like humans. It's used in various industries like healthcare and finance.	2	
		Artificial Intelligence is a field of computer science that focuses on creating intelligent machines. AI systems can perform tasks that typically require human intelligence, such as learning, problem-solving, and reasoning. These systems use algorithms, data, and computing power to simulate behavior. AI has transformed various industries and continues to revolutionize the way we live and work.	4	

Fig -5: Sample dataset

This dataset was created with attributes such as Question, Actual Answer, Student Answer, Obtained marks and Total score. The data was collected from mid-semester exam papers in the college. The dataset is designed to train a similarity model that compares student answers with the actual answer. The goal is to improve the accuracy of automatic answer evaluation systems by helping the model understand how closely a student's answer matches the actual answer.

Obtained marks= Cosine_similaritySBERT * Total score (2)

The score for each student's answer is calculated by multiplying the similarity score (from SBERT) with the total score. This shows how closely the student's answer matches the correct answer.

Phase 4: Answer Script Grading

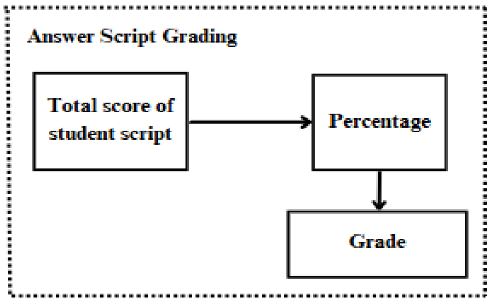

Fig -6: Grading

In this phase, the scores assigned to each individual answer are collected and given as input to the system. The system then adds up these scores to create a total score for the entire student script, reflecting the student's overall performance. After the total score is calculated, it is converted into a percentage, giving a clear way to see how the student performed compared to the total possible points. Once the percentage is determined, the final step is to give a grade based on grading conditions. The grade is usually on a scale like A, B, C, D, E and Fail that indicates the student's performance. This method ensures a fair and clear evaluation of the student's work, helping them understand how their answers affected the final grade. Finally, all the grades are saved in an Excel file, which is the final output. This Excel sheet lists the grades of all students; if a student is absent, it is marked as "Absent," and if a student is detained, it is marked as "Detained.

$$\text{Resultant Score} = \frac{\sum_{i=0}^{n} Obtained\ marks_i}{\sum_{i=0}^{n} Totalmarks_i} \quad (3)$$

- Obtained $marks_i$ is the score for the i^{th} answer, calculated based on similarity score and the total score of that question.
- Total $score_i$ is the maximum score for the i^{th} answer

4. RESULTS:

This Automated answer script evaluation system achieves an accuracy of 93.06% by utilizing both NLP and ML techniques to provide precise results that contribute to efficient performance evaluation.

Roll	Grade
1	B
2	Absent
3	E
4	Fail
5	A
6	A
7	C
8	E
9	B
10	Detained
11	E
12	D
13	B
14	B
15	B
16	C
17	A
18	A
19	A
20	A
21	Detained
22	B
23	Fail
24	Absent
25	B

Fig -7: Grade Allocation Report

The final output is saved in an Excel file, which contains two columns: Roll Number and Grade. Each row represents a student with their assigned grade, ranging from A to Fail, along with special statuses like Absent and Detained for students who either did not appear for the exam or were disqualified. This structured format allows for easy tracking of each student's performance and status.

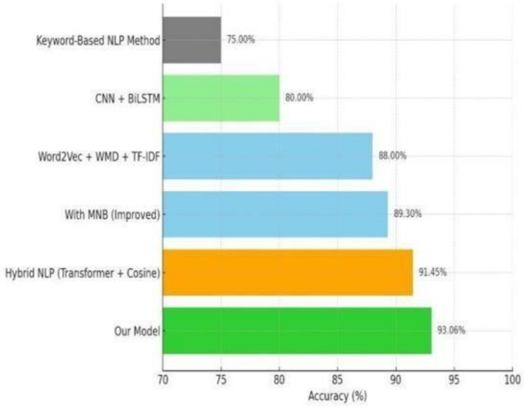

Fig -8: Comparsion of accuracies of Evaluation Models

The T5 (Text-To-Text Transfer Transformer) large model, developed by Google Research, was trained on the Colossal Clean Crawled Corpus (C4), which consists of over 750GB of cleaned English text—amounting to billions of tokens. The T5-large variant has 770 million parameters and is pre-trained on a multi-task framework, enabling it to perform tasks like summarization, question answering, and translation by simply prompting the task in natural language. When used in automated answer evaluation systems, T5 can generate high-quality answers or summaries that serve as gold standards for grading. These generated answers can then be compared to student responses using Sentence-BERT (SBERT), a modification of BERT that produces semantically rich sentence embeddings. Using cosine similarity, SBERT evaluates how close a student's answer is to the T5-generated reference, enabling accurate scoring. This combined T5-SBERT system helps the model better understand both the structure and the meaning of student answers, which leads to more accurate evaluation.

5. CONCLUSION:

The Automated Answer Script Evaluation project using ML and NLP efficiently evaluates open-ended, handwritten answers. By utilizing LLM APIs for text extraction, T5 for summarization and question-answering, and SBERT with cosine similarity for scoring, the system ensures accurate and unbiased assessment. It also supports manual entry of marks, handles absentee and detained students, and generates a structured Excel report with grades. This approach enhances evaluation speed, reduces human effort, and provides a systematic, data-driven assessment method for academic institutions.

FUTURE WORK:

Future work can focus on improving the system by adding image recognition to evaluate diagrams and graphs that the current model cannot handle. With computer vision techniques, the system can analyze visual content along with text, making it more effective for subjects that involve diagram-based answers. AI can also help grade mathematical equations, diagrams and flowcharts. These improvements will make the system more accurate and suitable for a wider range of subjects, increasing its overall efficiency in automated answer evaluation.

REFERENCES:

[1] Kulkarni, M., Adhav, G., Wadile, K., Chavan, R. and Deshmukh, V., 2024. Digital Handwritten Answer Sheet Evaluation System.

[2] Del Gobbo, E., Guarino, A., Cafarelli, B. and Grilli, L., 2023. GradeAid: a framework for automatic short answers grading in educational contexts design, implementation and evaluation. Knowledge and Information Systems, 65(10), pp.4295-4334

[3] Rahaman, M.A. and Mahmud, H., 2022. Automated evaluation of handwritten answer script using deep learning approach. Transactions on Machine Learning and Artificial Intelligence, 10(4).

[4] Rani, A.M., 2021. Automated explanatory answer evaluation using machine learning approach. Des Eng, pp.1181-1190.

[5] M. F. Bashir, H. Arshad, A. R. Javed, N. Kryvinska and S. S. Band, "Subjective Answers Evaluation Using Machine Learning and Natural Language Processing, " in IEEE Access, vol. 9, pp.158972-158983,2021,doi: 10.1109/ACCESS.2021.3130902. keywords:{Machine learning;Tokenization;Taskanalysis;Tagging;Vocabulary;Semantics;Data models;Subjective answer evaluation;big data;machine learning;natural language processing; word 2vec}

[5] Vij, S., Tayal, D. and Jain, A., 2020. A machine learning approach for automated evaluation of short answers using text similarity based on WordNet

graphs. Wireless Personal Communications, 111, pp.1271-1282.

[6] M. Han, X. Zhang, X. Yuan, J. Jiang, W. Yun, and C. Gao, "A survey on the techniques, applications, and performance of short text semantic similarity," Concurrency Comput., Pract. Exper., vol. 33, no. 5, p. e5971, Mar. 2021.

[7] J. Wang and Y. Dong, "Measurement of text similarity: A survey," Information, vol. 11, no. 9, p. 421, Aug. 2020.

[8] M. S. M. Patil and M. S. Patil, "Evaluating student descriptive answers using natural language processing," Int. J. Eng. Res. Technol., vol. 3, no. 3, pp. 1716–1718, 2014.

[9] P. Patil, S. Patil, V. Miniyar, and A. Bandal, "Subjective answer evaluation using machine learning," Int. J. Pure Appl. Math., vol. 118, no. 24, p. 113, 2018.

[10] J. Muangprathub, S. Kajornkasirat, and A. Wanichsombat, "Document plagiarism detection using a new concept similarity in formal concept analysis," J. Appl. Math., vol. 2021, pp. 1–10, Mar. 2021.

[12] X.HuandH.Xia,"Automatedassessment system for subjective questions based on LSI," in Proc. 3rd Int. Symp. Intell. Inf. Technol. Secur. Informat., Apr. 2010, pp. 250–254.

[13] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, "From word embeddings to document distances," in Proc. Int. Conf. Mach. Learn., 2015

[14] C. Xia, T. He, W. Li, Z. Qin, and Z. Zou, "Similarity analysis of law documents based on word 2 vec," inProc.IEEE19thInt.Conf.Softw.Qual., Rel. Secur. Companion (QRSC), Jul. 2019, pp. 354–357.

[15] H. Mittal and M. S. Devi, "Subjective evaluation: A comparison of several statistical techniques," Appl. Artif. Intell., vol. 32, no. 1, pp. 85–95, Jan. 2018.

[16] L. A. Cutrone and M. Chang, "Automarking: Automatic assessment of open questions," in Proc. 10th IEEE Int. Conf. Adv. Learn. Technol., Sousse, Tunisia, Jul. 2010, pp. 143–147.

[17] G.Srivastava, P. K. R. Maddikunta, and T. R. Gadekallu, "A two-stage text feature selection algorithm for improving text classification," China Med. Univ., Taiwan, Tech. Rep. 137, 2021.

[18] H. Mangassarian and H. Artail, "A general framework for subjective information extraction from unstructured English text," Data Knowl. Eng., vol. 62, no. 2, pp. 352–367, Aug. 2007.

[19] B. Oral, E. Emekligil, S. Arslan, and G. Eryiˇ git, "Information extraction from text intensive and visually rich banking documents," Inf. Process. Manag., vol. 57, no. 6, Nov. 2020, Art. no. 102361.

[20]V. Nandini and P. Uma Maheswari, "Automatic assessment of descriptive answers in online examination system using semantic relational features," J. Supercomput., vol. 76, no. 6, pp. 4430 4448,Jun. 2020.

[21]H. Tuan Nguyen, C. Tuan Nguyen, H. Oka, T. Ishioka, and M. Nakagawa, "Handwriting recognition and automatic scoring for descriptive answers in Japanese language.