Global Academy of Technology
Growing Ahead Of Time....
Autonomous Institute, Affiliated To VTU

Department of Artificial Intelligence & Machine Learning

Mini-Project Presentation – Review 1

Deep learning model for automated answer sheet evaluation.

Team Members:

Harshavardhan MG (1GA22AI015)

P Praveen Raj (1GA22AI031)

Project Guide:

Prof .Vani

Designation:

Assistant. Professor

Outline

- Introduction
- Literature Review
- Problem Statement
- Existing System
- Proposed System
- Objectives
- Architecture Diagram
- Technology Stack
- References

Introduction

- Grading answer sheets, especially those with a combination of textual and diagrammatic responses, is a complex and time-consuming process. Traditional grading methods are often slow and hectic, particularly when dealing with large volumes of exams.
- As student numbers grow, manually evaluating each answer sheet becomes increasingly difficult and time-intensive. There is a need for a more efficient and scalable solution that can handle diverse answer formats while ensuring accuracy and fairness.
- This project aims to develop an automated grading system that streamlines the evaluation process, reducing the workload for educators and providing faster, more consistent results. By automating the grading process, the system will improve the overall efficiency of assessments and offer timely feedback for better learning outcomes.

MINI PROJECT(22AMLP56)

TITLE: Deep learming model to evaluate answer sheets

Literature Review:									Bibliography:
Sl. No	Paper Title	Publication	Published Year	Objective	Dataset	Methodology	Result Discussion	Review	MLA;
1	An Enhanced Framework for Smart Automated Evaluations of Answer Scripts	International Journal of Innovative Technology (IJIT)	2021	To develop a hybrid system combining Optical Character Recognition (OCR) and Natural Language Processing (NLP) for automated grading, ensuring higher accuracy and efficiency.	Simulated handwritten answer sheets, using OCR for text extraction and NLP for semantic analysis. Word embedding techniques like Word2Vec and Bag of Words are explored	The framework involves pre-processing answer scripts, extracting keywords, analyzing sentence structures, and matching answers with model responses. Techniques include tokenization, stemming, lemmatization, and statistical measures like cosine similarity.	The framework improved accuracy and efficiency in grading. The hybrid use of OCR and NLP reduced manual efforts significantly, achieving an accuracy of 87% in test cases. Challenges included inconsistent handwriting and high computational cost.	The system demonstrated time efficiency compared to manual grading and showed improved reliability in subjective answer evaluation by leveraging deep learning techniques like CNNs and RNNs.	"An Enhanced Framework for Smart Automated Evaluations of Answer Scripts." International Journal of Innovative Technology, 2021.
2	A Study of Automated Evaluation of Student's Examination Paper Using OCR	International Journal of Innovative Technology (IJIT	2020	Improve handwriting recognition and grading accuracy using CNNs for feature extraction.	Scanned answer scripts processed using OCR to extract text, paired with teacher-prepared reference answers for validation.	Features are matched using NLP-based semantic comparison, focusing on keywords and sentence coherence. Stopword removal, stemming, and similarity metrics are key components.	Successfully digitized handwritten answer sheets and achieved over 80% accuracy in semantic matching. However, inconsistent handwriting and unstructured responses reduced reliability.	This method showed efficiency in grading but faced challenges with ambiguous handwriting and complex answer structures.	"A Study of Automated Evaluation of Student's Examination Paper Using OCR." International Journal of Innovative Technology, 2020.
3	Automated Grading for Handwritten Answersheets Using Convolutional Neural Networks (CNNs)	ResearchGate	2022	To make static data visualisations interactive using AR for better data exploration during presentations.	IAM Handwriting Database and sample student answer scripts.	Preprocessing includes noise elimination, segmentation, and feature extraction using CNN. Text is then evaluated with semantic models.	The use of CNNs resulted in high recognition accuracy (up to 90%) for clear handwriting but struggled with noisy datasets. Computational efficiency was a limitation.	High recognition accuracy was achieved for clear handwriting, but complex scripts posed challenges	Automated Grading for Handwritten Answersheets Using Convolutional Neural Networks (CNNs)." ResearchGate, 2022.

4	Answer Evaluation Using Machine Learning	International Journal of Innovative Technology (IJIT)	2021	Design an efficient system to evaluate descriptive answers using ML for semantic and structural analysis.	A small dataset of scanned handwritten responses paired with grading rubrics	Text vectorization (TF-IDF), keyword matching, and semantic similarity algorithms were applied to measure relevance.	Demonstrated effective semantic matching for descriptive answers with an accuracy of 85%. Struggled with long, subjective responses requiring deeper contextual understanding.	The system was reliable for shorter answers but required improvements for subjective or lengthy content	"Answer Evaluation Using Machine Learning." International Journal of Innovative Technology, 2021.
5	Answer Sheet Layout Analysis Using YOLOv5s-DC and MSER	ResearchGate	2023	Enhance OCR precision by analyzing the layout of answer sheets using YOLOv5s and MSER	Annotated answer sheet images with defined layout regions.	YOLOv5s identifies structural elements, and MSER extracts text from predefined areas. This is followed by OCR for text recognition.	Achieved precision above 90% for layout detection and text extraction. Integration of YOLOv5s and MSER significantly improved processing speed, but diverse formats posed challenges.	Achieved high accuracy in layout detection but required optimization for diverse formatting styles	"Answer Sheet Layout Analysis Using YOLOv5s-DC and MSER." ResearchGate, 2023.
6	Automated Script Evaluation Using NLP and Deep Learning	ResearchGate	2022	Automate the grading of answer sheets using OCR for text extraction and NLP for grading subjective answers.	Synthetic answer sheet dataset for training and evaluation.	Used OCR for text extraction and BERT-based NLP models for scoring. Grading is based on semantic similarity to model answers.	Scored high on consistency and reliability (accuracy 88%) for subjective grading using BERT models. Computational requirements and model training time were identified as limitations.	Demonstrated high accuracy for subjective grading but required extensive computational resources	"Automated Script Evaluation Using NLP and Deep Learning." ResearchGate, 2022.
7	Grading Descriptive Answer Scripts Using Deep Learning	International Journal of Innovative Technology (IJIT)	2021	Use deep learning models to evaluate descriptive answers, focusing on context comprehension and structure.	Custom answer sheet dataset processed using OCR for text extraction.	A combination of CNN for handwriting recognition and fuzzy logic for contextual evaluation.	Adapted well to diverse question formats with an accuracy of 83%. Challenges included processing ambiguous answers and dealing with biased datasets.	Showed adaptability to various question formats but faced difficulties with ambiguous content	"Grading Descriptive Answer Scripts Using Deep Learning." International Journal of Innovative Technology, 2021.

8	Indic Script Evaluation Using Deep Learning	ResearchGate	2023	Develop a system for evaluating Indic language scripts using OCR and deep learning models.	Indic language answer sheets from publicly available datasets and synthetic sources.	Language-specific preprocessing, OCR for text extraction, and deep learning for semantic grading.	Successfully handled linguistic complexities, achieving over 85% accuracy in semantic evaluation. The system required further optimization for minority languages.	Addressed linguistic complexities effectively but required further fine-tuning for less common languages	"Indic Script Evaluation Using Deep Learning." ResearchGate, 2023.
9	MCQ Evaluation Using Python OCR	ResearchGate	2022	Build an automated grading system for MCQs using Python-based OCR for text recognition.	Scanned MCQ answer sheets with predefined answer keys.	OCR was used to detect responses, which were compared with answer keys. The system flagged discrepancies for manual review.	Achieved efficient grading for structured MCQs with an accuracy of 92%. The method struggled with ambiguous marking patterns and handwritten responses.	Efficient for grading structured MCQs but required manual intervention for ambiguous cases	"MCQ Evaluation Using Python OCR." ResearchGate, 2022.
10	Automated Evaluation of Short Answers Using WordNet Graphs	ResearchGate	2022	Evaluate short answers using graph-based methods to measure semantic similarity.	Custom dataset of short textual responses with annotated scores	Graph-based similarity measures were applied using WordNet to compare student responses with model answers.	Semantic similarity techniques using WordNet graphs achieved accuracy of 87% for short answers. Scalability for longer or subjective answers was limited.	Effective for short answers but limited scalability for longer or more complex responses	Automated Evaluation of Short Answers Using WordNet Graphs." ResearchGate, 2022.

Problem Statement

“Grading answer sheets is a time-consuming and labor-intensive process. There can be an automated system that can provide consistent, accurate results reducing the grading workload.”

Existing System

- **Text and Handwriting Processing:** Existing solutions use OCR and deep learning models (e.g., CNNs) to extract and process handwritten text, achieving decent accuracy for clear handwriting. However, these systems struggle with inconsistent handwriting, unstructured responses, and noisy data, limiting their effectiveness for a wide range of answer formats.
- **Semantic and Diagram Evaluation:** Many systems integrate NLP techniques and machine learning models (like BERT) for evaluating subjective answers based on semantic similarity. Vision-based models, such as YOLO, are applied for diagram evaluation, but challenges arise in handling complex or diverse diagram structures and ensuring consistent grading for subjective responses.
- **Scalability and Resource Constraints:** While these solutions improve grading efficiency, they face issues with scalability, especially when handling large volumes of answer sheets. High computational costs and the inability to effectively grade mixed-content answers (text and diagrams) remain significant hurdles for widespread implementation.

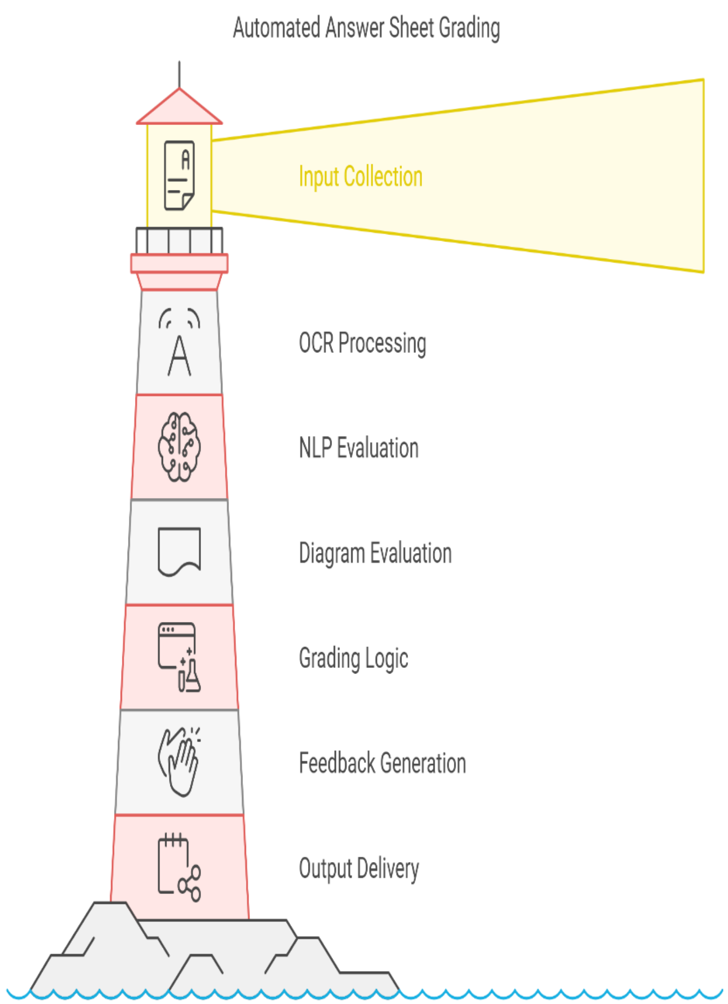
Proposed System

- The proposed system integrates OCR, NLP, and deep learning models to automate grading of answer sheets.
- **Text Grading:** OCR extracts text from scanned answer sheets, followed by NLP for semantic evaluation using models like BERT and GPT.
- **Diagram Evaluation:** CNNs and YOLO detect and grade diagrams based on reference images, providing partial credit for conceptual accuracy.
- **Reference Solution Input:** Educators manually upload reference solutions for each question, including diagrams and templates for subjective answers. The system compares student responses to these references to ensure accurate grading. Flexibility is provided by allowing educators to assign weights to specific sections or handle alternative correct answers.
- **Handwriting Standardization:** Preprocessing techniques standardize handwriting for improved recognition.
- **Mixed-Content Handling:** The system handles both text and diagram responses in a single answer sheet.
- **Real-Time Feedback:** Provides instant feedback, highlighting errors and areas for improvement.
- **Optimized Efficiency:** Uses advanced techniques to reduce computational costs, ensuring smooth performance.
- The system ensures a consistent, accurate, and efficient grading process for diverse exam formats.

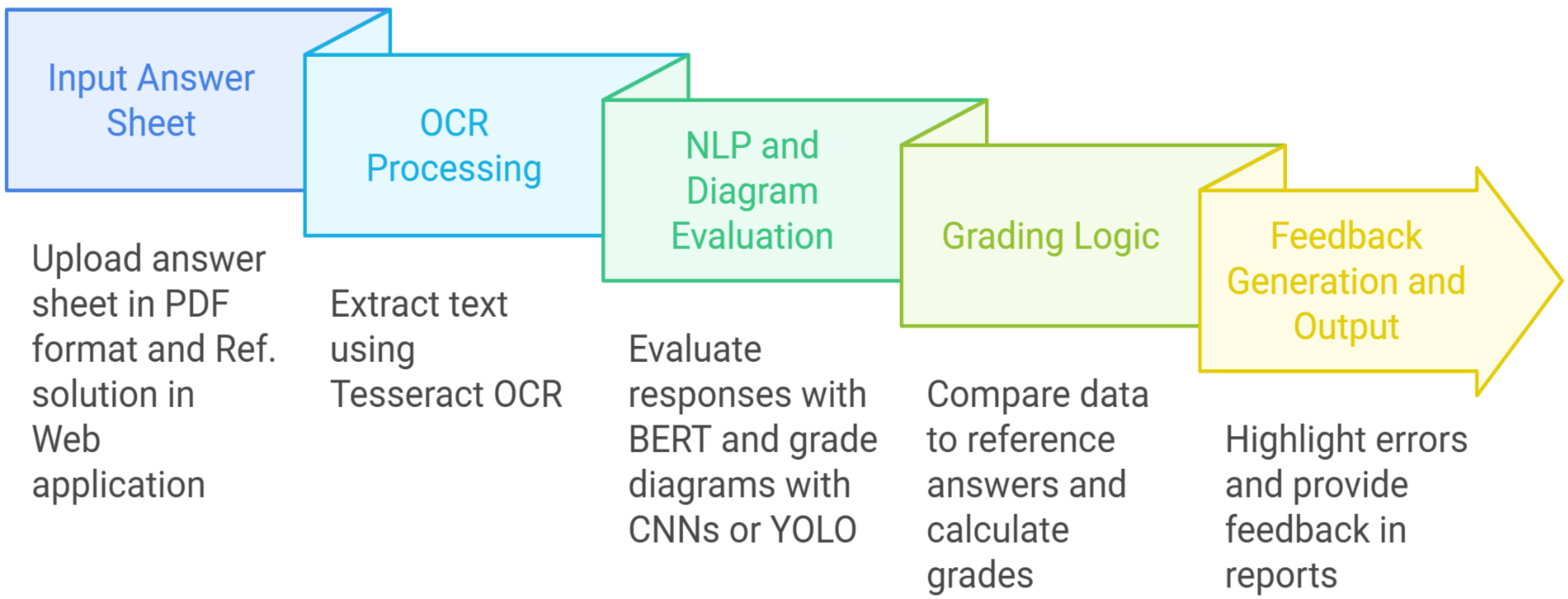
Objectives

1. To automate the grading of both text-based and diagrammatic answers, reducing manual effort and time.
2. To integrate OCR and NLP for accurate text extraction and semantic evaluation of student responses.
3. To apply deep learning models (e.g., CNNs, YOLO) for evaluating diagrams, ensuring accurate grading even for complex visual answers.
4. To handle mixed-content answer sheets, combining both text and diagrams in a single evaluation system.
5. To enable educators to upload reference solutions manually, including text templates and diagrams, while allowing flexibility in assigning weights and handling alternative correct answers.
6. To provide real-time, actionable feedback to students for continuous learning and improvement.
7. To design a scalable system capable of efficiently processing large volumes of answer sheets with minimal computational costs.

Architecture Diagram



Automated Answer Sheet Grading Workflow



Technology Stack

- **Programming Language:** Python (for model development, text processing, and integration)
- **Libraries and Frameworks:**
  - TensorFlow and PyTorch (for deep learning model development)
  - OpenCV (for image processing and diagram recognition)
  - Tesseract OCR (for text extraction from scanned answer sheets)
- **Databases:** SQL or NoSQL databases (for storing answer sheets, grading data, and results)

References

- Smith, R. (2007). "An overview of the Tesseract OCR engine." *Proceedings of the International Conference on Document Analysis and Recognition (ICDAR)*, 629-633.
- Tesseract OCR. (2021). "Tesseract: An Open Source OCR Engine.
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). "BERT: Pre-training of deep bidirectional transformers for language understanding." *Proceedings of NAACL-HLT 2019*, 4171-4186.
- Radford, A., et al. (2018). "Improving Language Understanding by Generative Pre-Training." *OpenAI Blog*.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). "You Only Look Once: Unified, Real-Time Object Detection." *Proceedings of CVPR 2016*, 779-788.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). "Deep learning." *Nature, 521*(7553), 436-444.
- Amazon Web Services. (2021). "What is AWS?"
- Google Cloud. (2021). "Cloud Computing Services."
- Chacon, S., & Straub, B. (2014). "Pro Git." *Apress*. Grinberg, M. (2018). "Flask Web Development: Developing Web Applications with Python." *O'Reilly Media*.
- Markuszewski, P. (2020). "FastAPI: The modern, fast (high-performance) web framework for building APIs with Python." *FastAPI Documentation*.