RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Mini Project Report On

Grade AI (An AI Tool that evaluates Answer Sheets)

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science & Engineering

By

Abhinav Sobi (U2103008)

Alan Joseph (U2103021)

Basil Eldho Joseph (U2103057)

Daniel Robin (U2103072)

Under the guidance of

Dr. Dhanya P.M.

Department of Computer Science & Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Affiliated to APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
May 2024

CERTIFICATE

This is to certify that the mini project report entitled ***"Grade AI"*** *is a bonafide record of the work done by* ***Abhinav Sobi (U2103008)***, ***Alan Joseph (U2103021)***, ***Basil Eldho Joseph (U2103057)***, ***Daniel Robin (U2103072)***, *submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Dr. Dhanya P.M.
Project Guide
Professor
Dept. of CSE
RSET

Mr. Harikrishnan M
Project Coordinator
Asst. Professor
Dept. of CSE
RSET

Dr. Preetha K. G.
Head of the Department
Professor
Dept. of CSE
RSET

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude towards Dr P. S. Sreejith, Principal of RSET, and Dr. Preetha K.G., Head of the Department of Computer Science and Engineering for providing me with the opportunity to undertake my mini project, "Grade AI".

I am highly indebted to my project coordinators, **Mr. Harikrishnan M**, Assistant Professor, Department of Computer Science and Engineering and **Ms. Sherine Sebastian**, Assistant Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed my pleasure and a moment of satisfaction for me to express my sincere gratitude to my project guide **Dr. Dhanya P.M.** for her patience and all the priceless advice and wisdom she has shared with me.

Last but not the least, I would like to express my sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Abhinav Sobi

Alan Joseph

Basil Eldho Joseph

Daniel Robin

Abstract

Grade AI is a web-based AI Answer Paper Evaluator which aims to revolutionize the education sector by automating the exam evaluation process. Institutions can effortlessly submit their exam papers, get them evaluated and view statistics and feedback within a single, user-friendly interface accessible from any device with internet connectivity.

The web application will allow institutions to create an account to view their exams history and results. New exam submissions can be created, following which the hand-written answer sheets are scanned and uploaded as PDF or image files. The answer key, based on which the answer sheets are to be evaluated, is also uploaded. The uploaded papers will be securely processed and stored on the server for evaluation.

The system will speed up the evaluation process by a huge factor when compared to the traditional evaluation process conducted by humans. It also helps to reduce biases and discrepancies which may occur due to multiple human evaluators. The answer papers will be evaluated based on the relevance, coherence, quality and semantic similarity of answers with respect to the answer key. Based on this, the system will automatically assign scores to the answer papers.

Exam results can be shared by the examination cell of the institution to the students, teachers and the public as required. The system also allows students to log-in and view their results, grades and rank. Overall, the AI Answer Paper Evaluator aims to redefine the evaluation process, reduce the workload on teachers, and provide more consistent and objective assessments of student performance, ultimately enhancing the quality of education.

Contents

List of Figures

List of Tables

List of Abbreviations

Acronym - Expansion

Chapter 1

Introduction

1.1 Background

Handwritten Exam paper evaluation is a hassle which troubles both educators and students alike. For the teachers or professors, exam evaluation is a tedious task which takes up a significant amount of their time and energy. The human biases arising from a diverse group of evaluators affects the quality of the evaluation. Students have to wait months before the exam results get published, this affects their future studies. Viewing the evaluation and applying for re-evaluation is also a hassle which takes up even more time and resources.
Hence the Education sector is in dire need of a swift, streamlined and accurate Exam Evaluation and Result Management process.

1.2 Problem Definition

To develop a platform for automatic grading of handwritten exam answer papers based on their semantic similarity, keyword matching, answer length and more.
The platform should also provide a seamless and user friendly result viewing and management experience.

1.3 Scope and Motivation

GradeAI has the scope to revolutionize the Educational sector. It aims to accelerates the text based exam evaluation.
The project is not suitable for similarity analysis of diagrams or graphs.
GradeAI evaluates any answer sheets consisting of English text only.

Motivation for the project are the countless unpleasant experiences associated with Exam evaluation which has plagued both students and educators throughout the years.
Teachers have to spent hours, even after midnight, to complete the evaluation.
Delay in result publishing has resulted in rejection of admissions for higher studies.
Students get marks which are less than they deserve due to human biases or errors.
Re-evaluation is a tedious and expensive process which takes up more time.

You may insert tables into your document using the given code:

Title 1	**Title 2**	**Title 3**
1	Content 1	Content 2
2	Content 3	Content 4

Table 1.1: Insert tables here

1.4 Objectives

1. GradeAI should be an easily accessible and user friendly platform for both educators and students.

2. Educator should be able to scan and upload answer keys and answer papers.

3. GradeAI should evaluate the answers of the students with respect to the answer key provided based on semantic similarity.

4. An exam administrator, who represents the educational institute, must be able to view and manage the data.

5 Students must be able to view their results, grade and rank for the exams they attended.

1.5 Challenges

Optical Character Recognition (OCR) technology is not advanced enough to perfectly extract text from poor hand-writings. OCR is also very resource intensive.

1.6 Assumptions

The availability and stability of the Google Gemini API is a critical factor in the project. The project assumes the API remains relatively stable in terms of functionality and response format.

The project assumes a certain level of OCR accuracy. Poor OCR results will render subsequent text comparison less meaningful, potentially requiring either manual intervention or a very flexible matching process tolerant of errors introduced by the OCR.

1.7 Societal / Industrial Relevance

Grade AI is relevant in the Education sector, from elementary schools to professional colleges. Any educational institution who conducts handwritten exams can use the services of the project. This section describes where the project can be applied, either for the society or the industry. Write the relevance applicable for the work.

1.8 Organization of the Report

This section should outline a roadmap of the contents in the report.

Chapter 2

Software Requirements Specification

Insert your SRS document here.

2.1 Introduction

Figure 2.1: Insert your images here, and provide necessary captions

2.2 Overall Description

2.3 External Interface Requirements

You can insert equations into your file using the below code:

$$a = b + c \tag{2.1}$$

$$= y - z \tag{2.2}$$

2.4 System Features

2.5 Other Nonfunctional Requirements

Chapter 3

System Architecture and Design

3.1 System Overview

This section gives an overview of the project. Detailed architecture diagram is expected in this section. The entire process has to be outlined in detail. You can write upto 2 pages.

3.2 Architectural Design

This section can use the design tools for a software product development like use case diagram, sequence diagram, ER diagram etc. No textual description is required. The titles for these diagrams should be chosen carefully so that they are self explanatory.

3.3 Dataset identified

This section describes the data source used in the project. Brief its properties and refer it to the appropriate location. Sample subsets of the dataset can be highlighted.

3.4 Proposed Methodology/Algorithms

This section describes in detail the methodologies or algorithms associated with your work. Algorithms should be written in appropriate format.

3.5 User Interface Design

The user interface design (wireframe designs) can be highlighted in this section. The figures titles should be in a chronological order and self explanatory.

3.6 Database Design

The detailed database design and its schema is expected in this section. The database used in the work can be mentioned here. The reason for choosing the database can be substantiated in this section.

3.7 Description of Implementation Strategies

"This section details the important implementation strategies used in your project. For example, (1) if you are doing a audio/ video capture, what is your python library and associated methods for doing that (2) if you are designing a CNN, how did you do that in the language which you are using for implementation (3) what are the different methods in the language you will use to evaluate your work. Small snippets of code can be supplemented to support your strategy." Libraries Used:

1. google.generativeai:

   This library provides tools for generative AI tasks, enabling users to create or train models for tasks such as image generation, text generation, and more.
   - **Methods:** Utilize pre-trained models or train custom models using techniques like GANs (Generative Adversarial Networks) or VAEs (Variational Autoencoders) for various generative tasks.

2. fastapi:

   FastAPI is a modern, fast (high-performance) web framework for building APIs with Python 3.7+.
   - **Methods:** Define API endpoints using FastAPI decorators, handle request and response validation using Pydantic models, and utilize asynchronous programming for high concurrency.

3. selenium:

Selenium is a web testing tool that can automate browser actions and interactions with web pages.
- **Methods:** Automate interactions with web pages, such as filling forms, clicking buttons, and scraping data from websites.

4. pydantic:

   Pydantic is a data validation and settings management library in Python.
   - **Methods:** Define Pydantic models to represent data structures, validate incoming data, and serialize data to and from JSON.

5. sqlalchemy:

   SQLAlchemy is a SQL toolkit and Object-Relational Mapping (ORM) library for Python.
   - **Methods:** Define database models using SQLAlchemy ORM, perform CRUD operations (Create, Read, Update, Delete), and execute raw SQL queries for database manipulation.

6. alembic:

   Alembic is a database migration tool for SQLAlchemy, providing a way to manage changes to a relational database schema over time.
   - **Methods:** Create and manage database migrations to version control database schema changes, allowing for easy upgrades and downgrades of database schemas.

3.8 Module Division

"This section describes the different modules involved in this project and a small description of the same is expected. This section ends with the information of which module is assigned to each project member."

1. User Interface Module:
   This module involves creation and implementation of the user interface for Grade AI.

It handles the design and functionality of features like user authentication, answer sheet uploading, and display of results.

2. OCR Integration Module:
   This module handles the integration of Optical Character Recognition (OCR), Handwritten Text Recognition (HTR) technology into the system. It processes uploaded answer sheets to extract text data to be used in the evaluation process.

3. Answer Correction Module:
   This module implements the algorithm for correcting answers based on extracted text data and a predefined answer key. It evaluates the similarity and correctness of answers.

4. Database Management Module:
   This module manages the storage and retrieval of answer sheets, corrected results, and associated metadata. It ensures data integrity and security.

1. User Interface Module -
2. OCR Integration Module -
3. Answer Correction Module -
4. Database Management Module -

3.9 Work Schedule - Gantt Chart

Outline the work schedule in the appropriate chart format.

Bibliography

[1] H. Garg and M. Dave, "Securing IoT Devices and SecurelyConnecting the Dots Using REST API and Middleware," 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Ghaziabad, India, 2019, pp. 1-6, doi: 10.1109/IoT-SIU.2019.8777334.

[2] M. Ebrahimi, Y. Chai, H. H. Zhang and H. Chen, "Heterogeneous Domain Adaptation With Adversarial Neural Representation Learning: Experiments on E-Commerce and Cybersecurity," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 2, pp. 1862-1875, 1 Feb. 2023, doi: 10.1109/TPAMI.2022.3163338.

[3] Reference 3

[4] Reference 4

[5] Reference 5

[6] Reference 6

[7] Reference 7

[8] Reference 8

[9] Reference 9

[10] Reference 10