Automatic Grading of Handwritten Answers

Harsh Jain[1], Mohd SherAli Shaikh[2], Ravi Shankar[3], Vinita Mishra[4]

[1,2,3] Student, Information Technology, VESIT, Mumbai, Maharashtra, India
[4]Professor, Information Technology, VESIT, Mumbai, Maharashtra, India

---***---

Abstract - In this digital world most of the activities are transitioning to an online medium which includes conducting exams online, but still pen and paper exams are given more priority when it comes to accreditation. During this pandemic we have seen that the traditional pen and paper exams at the exam centre were not possible and we were forced to use the online mode. In this online mode the answers can be submitted in two ways, first is digital MCQ form, and in second, the answers are written, scanned and submitted using a smart phone. In this paper, a solution to grading of papers of theory based subjects is obtained where Automatic Paper Grading will be performed using Natural Language Processing. We'll be using the OCR (Optical Character Recognition) algorithm for extracting the handwritten text from the papers and converting them into digital text. It will be graded by comparing the vector embeddings of the written answer and the answer provided by the teacher. This system will grade higher if the distance between the two answers in the vector form is small, i.e. , the similarity is higher.

Key Words: Machine Learning, Natural Language Processing, Optical Character Recognition, Vector Embeddings, Sentence Similarity

1. INTRODUCTION

For our project, we have tried to identify one of the most pressing problems in the current education system and tried to come up with a solution that will help the professors and other staff of educational institutes in general. Today, with the growing number of online classes and modes of education, there is a shortage of staff that can assess the exams written by students. Speeding up the evaluation remains as the major bottleneck for enhancing the throughput of instructors. Teachers spend a lot of their valuable time on correcting hundreds of answer papers, time which can be better spent on other work like projects, research or generally helping students. This technique is significant since MCQ examinations cannot always be used to assess a student's grasp of a subject. Our system will automatically grade handwritten papers without manual supervision of any kind and with a lower rate of error than normal. The system also provides a full evaluation of the student's performance on the test, allowing the teacher to stay up to speed on the student's strengths and weaknesses and help them develop. In the current situation of the world, this tool will save a lot of valuable time and effort which can be directed towards something more productive.

2. OBJECTIVES

- The primary objective of this system is the extraction of text from a handwritten paper by a student, followed by pre-processing by the system.
- This system will quickly generate the result by comparing the student's answer, with one or more correct answers.
- This system will make use of NLP and image processing that will help in high accuracy. [4]
- To design a system that will require a minimal amount of time to provide an evaluation while not compromising on the accuracy.
- To provide a detailed assessment report of the student's performance in the test to the respective student.[1]

3. LITERATURE SURVEY

We studied multiple papers and their findings are being summarised in this section(Fig 1). This section illustrates papers studied before and during the development of the project. The papers helped in gaining insight into existing solutions, possible ways to optimize algorithms and facilitate the selection of algorithms based on their performance. Figure 1 shows a comparison between all the papers that were referred to get a contrast between existing solutions of similar nature.

https://core.ac.uk/download/pdf/84280126.pdf	https://ieeexplore.ieee.org/document/4766677.	https://www.academia.edu/37117327/Handwritten_Short_Answer_Evaluation_System_HSAES	https://ieeexplore.ieee.org/document/8473170	https://ieeexplore.ieee.org/abstract/document/8583774	https://arxiv.org/pdf/1906.03767.pdf
This paper is focused only on handwritten numerical answers.	This paper focuses on essay grading systems	Natural Language Processing is used to extract the keywords.	Using ontology, extraction of words and their synonyms related to the domain is done	Integration of Optical Handwriting Recognition (OHR) and Automated Essay Scoring (AES)/grading.	Use of two segmentation networks based on DeepLabv3+ to locate the answer areas. Then, they used the character recognition part to recognize student's answers.
CNN	NLP	OCR, NLP	NLP, Semantic Analysis	Recurrent Neural Networks (RNN),ANN,MDLSTM,GloVe algo	CNN, DeepLabv3+, CRNN network`

Fig. 1. Literature Survey Comparison

4. TECHNICAL DEFINITIONS

This section acts as a preface for all the technologies that would be referred in the following section (See section V)

4.1 GOOGLE CLOUD VISION API

Vision API offers powerful pre-trained machine learning models through REST and RPC APIs. Assign labels to images and quickly classify them into millions of predefined categories. Detect objects and faces, read printed and handwritten text, and build valuable metadata into your image catalogue. We will detect handwritten text from an image using Google Vision API in python. We can use any image file with handwritten text or we can create our own sample image with own handwritten text.

With the Vision API, we can conduct OCR operations with minimal lines of code. In comparison to typical Python or R OCR libraries, it particularly performs well for handwritten text extraction(Fig 2). We chose Google Vision API because it gave us the best level of confidence and industry leading accuracy.

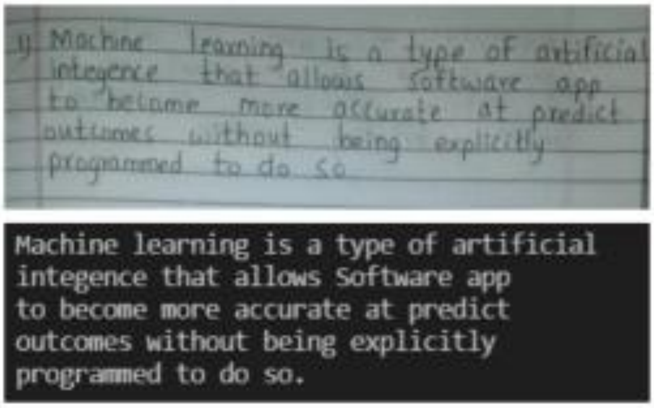


Fig. 2. Conversion of handwritten text to digital text.

4.2 BERT

BERT (Bidirectional Encoder Representations from Transformers) is a recent paper published by researchers at Google AI Language. It has caused a stir in the Machine Learning community by presenting state-of-the-art results in a wide variety of NLP[4] tasks. BERT's key technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modelling. This is in contrast to previous efforts which looked at a text sequence either from left to right or combined left-to-right and right-to-left training. The paper's results show that a language model which is bidirectionally trained can have a deeper sense of language context and flow than single-direction language models. BERT makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text. BERT has the ability to embed the meaning of words into densely packed vectors. We call them dense vectors because every value within the vector has a value and has a reason for being that value — this is in contrast to sparse vectors, such as one-hot encoded vectors where the majority of values are 0. BERT is great at creating these dense vectors, and each encoder layer (there are several) outputs a set of dense vectors.

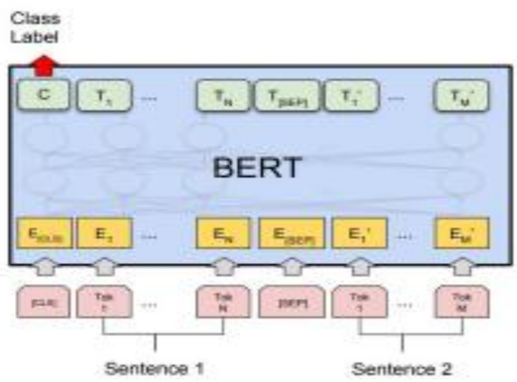


Fig. 3. BERT

For BERT base, this will be a vector containing 768. Those 768 values contain our numerical representation of a single token — which we can use as contextual word embeddings. Because there is one of these vectors for representing each token (output by each encoder), we are actually looking at a tensor of size 768 by the number of tokens. We can take these tensors — and transform them to create semantic representations of the input sequence.

4.3 WORD MOVER DISTANCE

The word mover's distance (WMD) is a fundamental technique for measuring the similarity of two documents. As the crux of WMD, it can take advantage of the underlying geometry of the word space by employing an optimal transport formulation. The original study on WMD reported that WMD outperforms classical baselines such as bag-of-words (BOW) and TF-IDF [3] by significant margins in various datasets. The sentences have no words in common, but by matching the relevant words, WMD is able to accurately measure the (dis)similarity between the two sentences. The method also uses the bag-of-words representation of the documents (simply put, the word's frequencies in the documents), noted as d in the figure below. The intuition behind the method is that we find the minimum "traveling distance" between documents, in other words the most efficient way to "move" the distribution of document 1 to the distribution of document.

4.4 COSINE SIMILARITY

Cosine similarity is one of the metrics to measure the text similarity between two documents irrespective of their size in Natural language Processing. A word is represented into a vector form. The text documents are represented in n-dimensional vector space. Mathematically, the Cosine similarity metric measures the cosine of the angle between two n-dimensional vectors projected in a multi-dimensional space. The Cosine similarity of two documents will range from 0 to 1. If the Cosine similarity score is 1, it means two vectors have the same orientation. The value closer to 0 indicates that the two documents have less similarity. The mathematical equation of Cosine similarity between two non-zero vectors is (Figure 4):

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Fig. 4. Cosine Similarity

4.5 WORD2VEC

Word2Vec is one of the most popular technique to learn word embeddings using shallow neural network. It was developed by Tomas Mikolov in 2013 at Google. Word2Vec is a method to construct such an embedding. It can be obtained using two methods (both involving Neural Networks): Skip Gram and Common Bag Of Words (CBOW). Word2vec is not a singular algorithm, rather, it is a family of model architectures and optimizations that can be used to learn word embeddings from large datasets. Embeddings learned through word2vec have proven to be successful on a variety of downstream natural language processing tasks.

5. PROPOSED SOLUTION

The solution proposed (Fig 5) is a system where the student can upload their scanned answer sheets in a PDF format on the portal which will be converted into separate images using python library poppler each corresponding to a page in PDF. We then use Google Vision API for obtaining the handwritten answers from the student's scanned images using text detection. This obtained answer is then compared to the ideal answer for that question, which is provided by the teacher in a database. Before comparing the answers, we apply the BERT algorithm for converting both answers in their respective vector forms. We then use the cosine similarity algorithm for measuring the similarity between the two vectors [3].

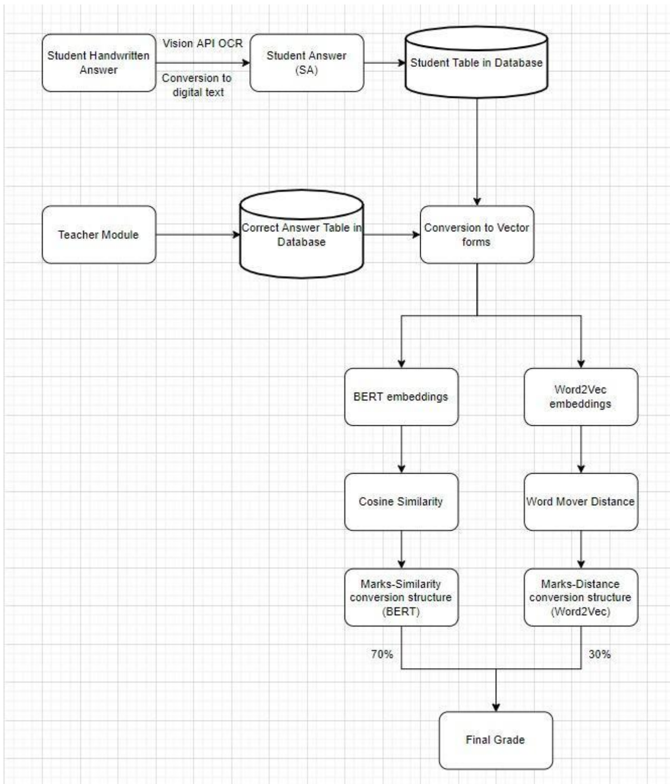


Fig. 5. Workflow of the System

This similarity is then compared with our marks-similarity conversion structure(Fig 6) to assign marks which will attribute to 70 percent of the total marks. This structure is the result of our analysis on data collected from students. The Remaining marks are calculated using WordMoverDistance in this the answers of student and teacher are pre-processed and sent to Word2Vec model which converts it into vector forms.

Similarity	Marks
0.75-1	100%
0.65-0.74	80%
0.55-0.64	65%
0.4-0.54	50%
Less than 0.4	0%

Fig. 6. Marks-Similarity Conversion Structure

We then use WordMoverDistance which returns distance. This distance indicate the similarity between answers. This distance is then compared with marks-distance conversion structure (Fig 7) to assign marks which will attribute to 30 percent of the total marks.

Distance	Marks
0-1	100%
1-2.5	80%
2.5-4	65%
4-5	50%
Greater than 5	0%

Fig. 7. Marks-Distance Conversion Structure

According to our observation and research, the combination of the similarity measure given by BERT, and the dissimilarity measure given by WMD, gives us the most optimal solution. The final sum of marks is then displayed to the student on the same portal.

The stakeholders of the system are basically every type of user that would interact with the system in its entirety. The stakeholders for the proposed system are:

- Students:

1) The user interacting with the system without any administrative powers.

2) Functionalities

i) Upload scanned documents on portal.

ii) Logging into the system.

iii) View final grade of the auto-assessed answer sheet

- Teachers:

1) The users interacting with the system without any administrative powers

2) Functionalities:

i) Logging in to the system.

ii) Add the ideal answers for all the questions.

iii) Teachers can view students marks.

- Administrator:

1) The highest authoritative user of the system.

2) Functionalities:

i) Maintenance of database.

ii) Maintenance of the portal.

iii) Add, remove, update any answers inputted by the teachers.

The solution is divided into 4 phases:

5.1) Data Collection and Fetching.

5.2) Converting answers into vector embeddings.

5.3) Grading the paper.

5.4) Application Interface.

5.1 DATA COLLECTION AND FETCHING

We have created a SQL database (as shown in Figure 8) which allows the students as well as the teachers to authenticate and login. This database is the single point of all the data collection and fetching in this project. This database has four main tables: [4]

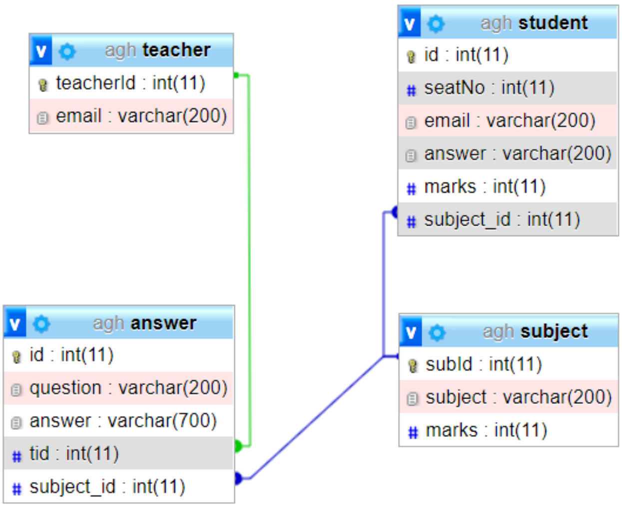


Fig. 8. Database Schema

- Answer: This table holds all the answers that the teacher will input via the portal, These are the ideal answers to all the questions and the student's answers will be compared to these for similarity and then graded appropriately. These answers will be categorized by the subject of the paper.

- Student: This table holds all the credentials of the students allowing them to authenticate and login, and also the once the answers are converted by the API, they are uploaded to this table before any further processing. The final grade of the student is also stored in this table.

- Subject: This table defines the subject of the paper and also used for referencing the subject in the student table.

- Teacher: This table holds the credentials of the teachers allowing them to authenticate and login.

5.2 CONVERTING ANSWERS INTO VECTOR EMBEDDINGS

- The processes to receive the student's answer in handwritten form and then convert it to digital text are as follows.

- The student will first login on the portal with his/her ID, choose the subject and then upload the PDF of the answer sheet.

- After uploading the PDF, the system will first breakdown the PDF into singular images using pdf2image, python's inbuilt function.

- Now each single image will only have one answer on it, which will be fed into the Vision API code in the same order.

- The output received from the Vision API (the converted digital text) is then systematically stored in an array.

- At this point, the teacher's answers are also retrieved from the database and those answers are stored in another array.

- Both the arrays are then encoded using BERT's model and Word2vec model. This gives us the vector form of all the answers. Now we can apply a similarity algorithm to compare these vectors.

5.3 GRADING THE PAPER

- The first step is applying the cosine similarity[3] on the vector embeddings that we just got from the BERT model.

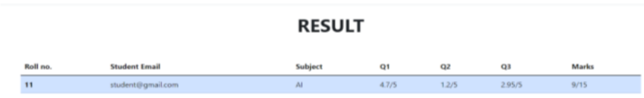
- The output of this will provide us with a number between 0 and 1, where 1 would mean the answers are very similar and 0 would mean the answers are extremely dissimilar. •The second step is applying the WordMoverDistance on the vector embeddings that we just got from the Word2vec model.

- The output of this will provide us with a number which is nothing but distance between two answers,higher the distance lower the similarity and lower the distance higher the similarity .

- The first step contributes 70 percent of total marks and the second step contributes 30 percent which then combined gives the total marks.

- After we have both the measures, we use the marks similarity conversion structure. We built this structure by creating a data set of many students answers to a same set of questions. We applied our system to these answers and found the optimal brackets for assigning various marks.

- The student can view his/her grade on the same portal. (Fig 9)



RESULT

Roll no.	Student Email	Subject	Q1	Q2	Q3	Marks
11	student@gmail.com	AI	4.7/5	1.2/5	2.95/5	9/15

Fig. 9. Final Result of a student

5.4 APPLICATION INTERFACE

We have created a website portal which serves as the interface that the teachers and students will use to interact with the system. The portal has been made secure by an authentication system. Both, students and teachers will have to login before doing anything else. We can break down the interface into two parts- one for the teacher and one for the student.

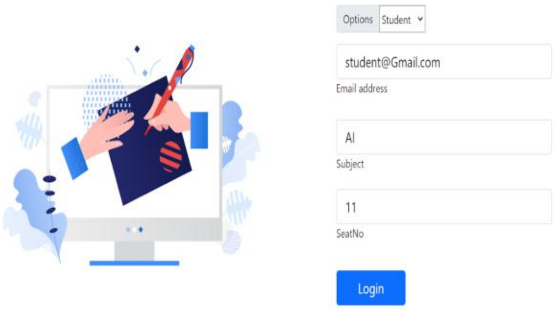


Fig. 10. Login Page

- Teacher: After logging in(Fig 10), the teacher will be allowed to choose a subject, choose the number of questions for the test, and then type in the questions and answers for that test.

- Student: After logging in(Fig 10), the student will be allowed to upload the scanned copy of his answer sheet in a PDF format(Fig 11). After this the student will see an option to grade the paper(Fig 12). Upon clicking this button, the student will receive the grade for his/her paper.

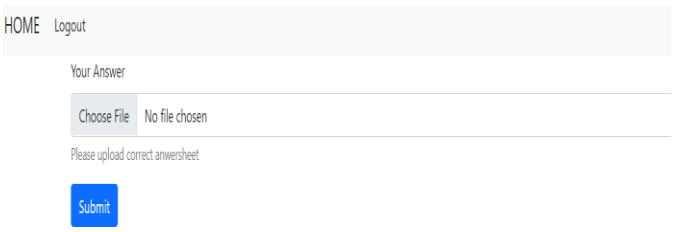


Fig. 11. Answer sheet upload page

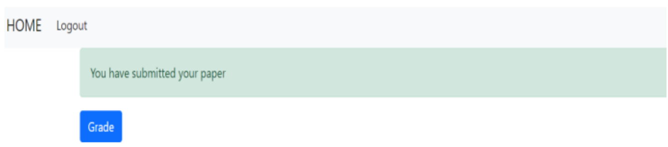


Fig. 12. Grading page.

6. INFERENCES

We were able to convert the students' handwritten answers into digital text using Vision API. Both the student's and teacher's answers were stored in the database along with the questions and the subject of the paper. The authentication system on the interface also ensured the security of the system, protecting it from any external threats or corruption. After integrating BERT and Cosine similarity [3] with WMD and Word2Vec, the marks-similarity conversion structure gave us optimal marks according to the final similarity rate. We successfully managed to get accurate marks for the students answers to the questions.

We have compared the accuracy of the marks given by our system , against the way an actual teacher would grade that same paper. This will give us an idea as to how close our system is, to an actual teacher. [6]

According to the graph (as shown in Figure 13), we can safely assume that the marks given by the system are extremely close to the marks given by an actual teacher. This allows us to believe that our system works successfully.

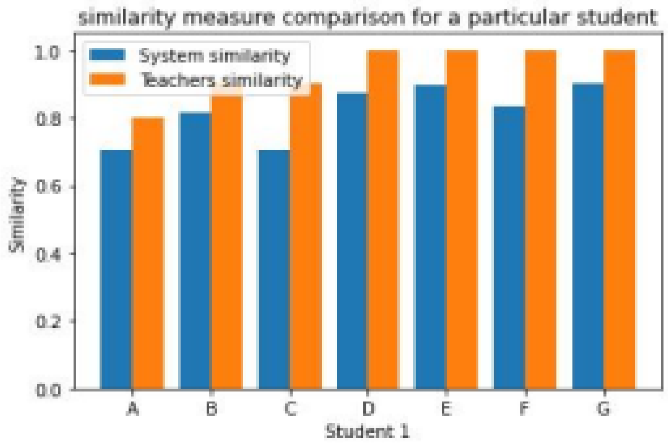


Fig. 13. System-Teacher Grade Comparison.

6. CONCLUSION AND FUTURE SCOPE

The system is capable of taking the answer sheet from the student and with the help of OCR and NLP[4], the system will analyse each answer. The similarity between the Student's and Teacher's answer. Based on these similarities, the marks will be calculated and assigned to the student. Although the system implemented is an optimal solution to the current established problem, there is room for improving the performance and managing multi-user interaction onto the system. Following are the future goals:

- Deploy the system on cloud platforms to ensure smoother processing of data.

- Build a mobile app for better and simple use of the system.

- Implement multi-user interaction successfully and test the entire system for security vulnerabilities.

- Getting rid of current constraints like mathematical equations and diagrams which cannot be graded with the current system.

- Further improve the accuracy of the grading system.

REFERENCES

[1] Mark Brown,"Automated Grading of Handwritten Numerical An swers",Computer Engineering Program California Polytechnic State University San Luis Obispo, CA,June 2017

[2] Siddhartha Ghosh, Sameen S. Fatima,"Design of an Automated Essay Grading (AEG) system in Indian context",published at TENCON 2008 - 2008 IEEE Region 10 Conference

[3] Sijimol P J, Surekha Mariam Varghese ,"Handwritten Short An swer Evaluation System (HSAES)",Department of Computer Science and Engineering, Mar Athanasius College of Engineering, Kothaman galam,Kerala, India,International Journal of Inventive Engineering and Sciences (IJSRST).

[4] Amit Rokade, Bhushan Patil, Sana Rajani, Surabhi Revandkar, Rajashree Shedge,"Automated Grading System Using Natural Language Process ing",2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT) ,IEEE.

[5] Annapurna Sharma, Dinesh Babu Jayagopi,"Automated Grading of Handwritten Essays",2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR),IEEE.

[6] Xiaoshuo Li , Tiezhu Yue , Xuanping Huang , Zhe Yang† , Gang Xu‡,"BAGS: An automatic homework grading system using the pictures taken by smart phones" ,Department of Physics, Tsinghua University, Beijing, China ‡School of Life Sciences, Tsinghua University, Beijing, China.