Assistive System for Answer Sheet Evaluation

1st Vijay M G
dept. of Information science and engineering.
Sai Vidya Institute of Technology
Bangalore, India
vijaymg.18is@saividya.ac.in

2nd Anoop R
dept. of Information science and engineering.
Sai Vidya Institute of Technology
Bangalore, India
anoopr.18is@saividya.ac.in

3rd Monish M
dept. of Information science and engineering.
Sai Vidya Institute of Technology
Bangalore, India
monishm.18is@saividya.ac.in

4th S Harsha Surya Datta
dept. of Information science and engineering.
Sai Vidya Institute of Technology
Bangalore, India
harshassuryadatta.18is@saividya.ac.in

Mrs. Nagarathna C
Assitant Professor
dept. of Information science and engineering.
Sai Vidya Institute of Technology
Bangalore, India

***Abstract*— Manual labor is difficult and exhausting when it comes to subjective paper evaluation. When employing Artificial Intelligence to analyses subjective papers, insufficient understanding and acceptance of data are major obstacles (AI). Several attempts have been made to use computer science to score students' answers. To accomplish this duty, the majority of the effort relies on traditional counts or specific terms. Additionally, there is a scarcity of curated data sets. This study provides a novel approach for automatically evaluating descriptive answers that incorporates multiple machine learning, natural language processing techniques, and tools. Answers are evaluated using solution statements and keywords, and a machine learning model is developed to predict answer grades.**

Keywords – Subjective answer evaluation, machine learning, natural language processing.

I. A. INTRODUCTION

In each educational institution or university, the examination procedure is one of the most important performance measurement criteria. Grading pupils and assessing their performance is an essential part of the evaluation process. There is a distinction between theoretical and practical disciplines. Answer sheets are evaluated in a traditional and manual manner[1]. Many people still listen to it. Universities and educational establishments The logistics of processing answer sheets by hand are numerous. as well as administrative procedures.

There is a need for automation in answer assessment systems in our current day, when everything in the world is heading towards automation. Multiple-Choice Questions are currently reviewed using OMR technology, but no other types of answers have yet been evaluated. (i.e., technical and non-technical parts of mathematics, algorithms, and programmes, subjective replies, accounting, and so on). In one form or another, much work has been done on the topic of subjective answers evaluation, such as measuring similarity between different texts, words, and even documents, finding the context behind the text and mapping it with the solution's context, counting noun-noun pairs in documents, matching keywords in answers, and so on. However, issues such as Tf-Idf losing its semantic context [6], a lack of hyper-parameter tweaking [7], expensive training [8, and the need for richer datasets [5] persist.

The most difficult and time-consuming activity in the educational system is assessing students using descriptive replies. Descriptive response assessment approaches have been studied for a long time, and numerous algorithms have been suggested and applied as a result. The majority of existing approaches focus on grammar, vocabulary, and content size rather than the semantic meaning of the text provided by the learner. The goal of this study was to develop a machine-learning model for analysing text semantics-based evaluation procedures for descriptive replies in exams.

There is a need for automation in answer assessment systems in our current day, when everything in the world is heading towards automation. Multiple-Choice Questions are currently reviewed using OMR technology, but no other types of answers have yet been evaluated [3]. (i.e., technical and non-technical parts of mathematics, algorithms, and programmes, subjective replies, accounting, and so on).

B. DESCRIPTION OF THE PROJECT

The existing system for checking abstract papers is inefficient. Assessing Subjective Answers is a simple task to complete. When an individual evaluates something, the nature of the evaluation may shift in tandem with the individual's emotions. To address this problem, we present a framework that combines Artificial Intelligence (AI) and Natural language processing (NLP)[2]. Our project aims to develop an associate that measures the right response sheet before handing it on to a human evaluator, resulting in a more efficient and precise evaluation interaction. The midway assessment involves Artificial Intelligence, in which a composite calculation is used to combine the results of various Machine Learning (ML) calculations to produce a forecast with more precision than its parts.

C. SCOPE OF PROJECT

1.Improvement in control and performance

The system is developed to cope up with current issues and problems of the automated evaluation of answer scripts.

2.Accuracy of the evaluated marks

In comparison with the manual evaluation deviance should be less than 10%.

3.Easy to use

A user uploads the answer scripts into the website and the rest will be taken care by the system making it easy to use.

4.Efficiency and speed of evaluation

Automated system will be capable of evaluating at least 10 subjective answers in and under 15 minutes.

D. CHALLENGES FACED BY AUTOMATIC EVALUATION SYSTEMS.

Traditional Manual Evaluation Methodologies Face Challenges Time-consuming and repeated operation, which is occasionally inaccurate and inefficient. Maintenance and Storage, the evaluators can influence grading.
It is not possible to do optimal data pre-processing and semantic analysis. As a result, machine learning models' capabilities are reduced. In some domains, the underlying method utilized in existing script evaluation systems is inefficient, which reduces the system's overall reliability.

II. BACKGROUND AND LITERATURE SURVEY

As previously stated, assessing subjective responses is not a novel concept; it has been studied for nearly two decades. Big data Natural Language Processing, Latent Semantic Analysis, Bayes theory, K-nearest classifier, and even formal techniques like Formal Concept Analysis have been used to solve this problem. Statistical, Information Extraction, and Full Natural Language Processing are the three main types.

Another issue that arises when utilising machine learning algorithms is the data that is provided to it. The set of features extracted from a given dataset must be carefully considered because it has a significant impact on the accuracy of the final output once it has been submitted to the classifiers. As a result, pre-processing is a critical component of our system. Text classification and feature extraction are difficult processes. We will look at, some text categorization techniques, such as Latent semantic analysis (LSA), were tested and found to improve the accuracy of the final forecast by a significant margin.

A.STATISTICAL TECHNIQUE

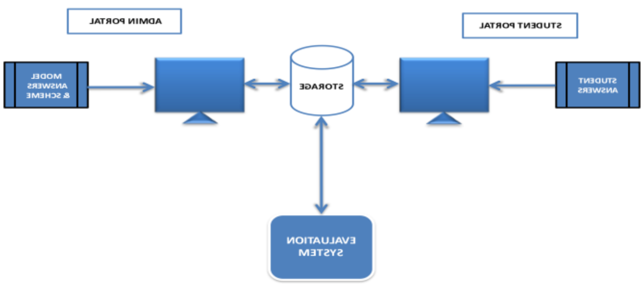
It is based on keyword matching and is considered inadequate because it cannot handle issues such as synonyms or context. This technique has been used in a number of studies on subjective paper rating.

1. INFORMATION EXTRACTION (IE) TECHNIQUE.
   Information extraction approaches rely on extracting a structure or pattern from the text in order to break it down into concepts and their connections [11]. The dependencies discovered to have a substantial influence in generating scores must be validated by a domain expert.

2. TOKENIZATION
   We use tokenization for the process of breaking down large amounts of data into smaller chunks, such as paragraphs, sentences, words, and characters [16].
   When dealing with natural language, tokenization is necessary since each word must be evaluated separately to determine its underlying meaning. We use white spaces and period marks to tokenize data into sentences and words in this project. One of the first steps in natural language processing is tokenization.

3. STOPWORD REMOVAL
   Natural Language has a large vocabulary, and most elements, such as 'the,' 'in,' 'on,' and 'is,' are there for human comprehension. These terms have little to no meaning in most machine learning tasks and may even obstruct the process by assisting the model in being trained on random data. Every language has certain well-known stop words, which are typically eliminated from the corpus in order to make the dataset denser and more distinctive. Schofield et al. [14] contend that the use of stopword removal is shallow, and that deleting stopwords beyond broad phrases has minimal impact on subject inference. According to agatayli and elebi [15], the effect of stopword removal has a minor impact on the real outcomes. It should be emphasised, however, that in order to strengthen the machine learning model, frequent terms of minor value should be deleted.
4. LEMMATIZATION
   Natural language words come in a variety of tense types. The terms 'go,' 'going,' and 'went,' for example, are all related to the core word 'go,' but have different forms. The process of lemmatization is the reduction of all words in a

dataset to their root forms. To tie words to their



lemma, often called the root, lemmatization requires a complete dictionary of the words. It also makes use of some speed information to link the terms to their dictionary roots. To categorise Italian text, Camastra and Razi [12] employed Lemmatization and support vector machines.

5. STEMMING
Stemming is a method of reducing words to their stems. It is founded on the assumption that every language has some type of formal grammar, and that the vocabulary is produced by remembering those rules. We can reduce all the comparable words back to their stems by deleting the suffixes that differentiate them using the same principles. There are various stemming algorithms forever in every language, such as Potter's algorithm for English word stemming. Jabbar et al. [9] discusses various stemming algorithms used to stem textual data.

6. BAG OF WORDS
Bag of Words is a basic technique that requires expressing the textual data's vocabulary as a vector. The index number in that vector represents either the count or the specific word at that index in the text. Bow keeps track of word frequencies but ignores their context. A one-hot vector is an example of BoW.Aryal et al. [11] employed the bag-of-words (BoW) vector representation to compare the similarity of two documents for each keyword that appeared in both.

7. COSINE SIMILARITY
Cosine similarity is a measure of similarity that measures the cosine of the angle between two non-zero vectors in an inner product space. The cosine angle between two vectors is calculated, with a value between 0 and 1,1 indicating a perfect match. Park et al. [10] proposed using cosine similarity to improve the performance of traditional classifiers as MNB, SVM, and CNN.

$$\textit{Cosine-similarity}\ (A, B) = \frac{(A, B)}{\|A\|.\|B\|}$$

III. SYSTEM DESIGN

Our goal is to determine which modules should be included in the system in order to efficiently meet all of the system's requirements. All of the modules' specifications, their interactions with other modules, and the expected output from each module will be included in the design. A description of the software architecture is the result of the design process.

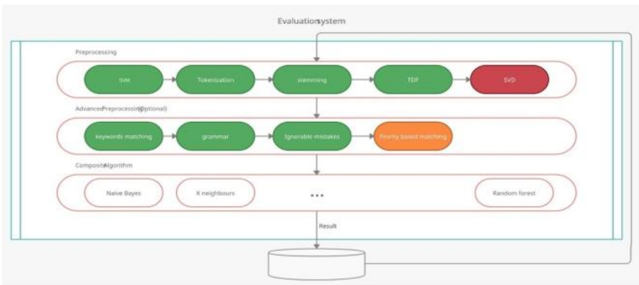
A. System Architecture

This figure shows the overview of the system architecture which has 4 components

1. Student - Used to take up the examination and view the results
2. Evaluator - Used to initiate examination by providing Question papers, Model answers and scheme.
3. Storage – Used to store the data of all the components
4. Evaluation System – Used to evaluate given examination.

It Accepts the answer scripts submitted from the student in the format of JPG or PDF. The model will be trained with the scheme of evaluation provided by the admin. The answer scripts will be evaluated and the result will be stored in the Database. Students then can view the results which is stored in the Database.

B. DateFlow Diagram



The evaluation system consists of:

1. DATA PREPROCESSING:

Step 1: Convolutional Neural Network
Step 2: Cleaning the Data and Tokenization
Step3: Spelling Corrector (minimum edit distance)
Step 4: Stemming

2. ADVANCED PREPROCESSING:
After all the preprocessing steps the features give to the Algorithm:

i. Using term frequency filtered bag of words.
ii. LSA – bag of concepts (only for large answers).
iii. Relevant words and bi grams.
iv. Key similarity score.

3.COMPOSITE ALGORITHM: -
Used to combine the results of multiple machine learning algorithms to achieve a combined accuracy higher than that of its parts. Composite algorithms are a type of algorithm that allows you to make more complex ones out of simpler ones. These could be more complicated algorithms that desire to use more atomic operations than others.

IV. IMPLEMENTATIONS

The automatic conversion of text in a picture into letter codes that may be used in computer and text processing applications is known as handwriting recognition.

Step 1: Using a CNN architecture, create a digit(0-9) Plus A-Z characters classifier.

A) Choose a Dataset
B) Prepare Dataset for Training: This involve assigning paths and creating categories(labels), resizing our images.
C) Create Training Data
D) Shuffle the Dataset
E) Assigning Labels and Features
F) Normalising X and converting labels to categorical data
G) Split X and Y for use in CNN
H) Define, compile and train the CNN Model
I) Accuracy and Score of models.

Step 2: For the handwritten word image, use character segmentation.

Step 3: Sort each segmented letter into a category, and then find the completed word in the image.

V. RESULTS

Figure 1. Student Login

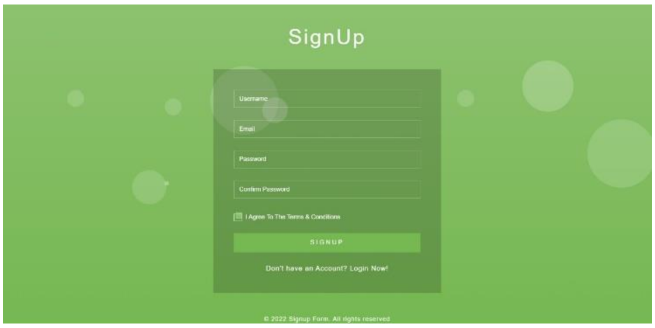


Figure 2. Student registration

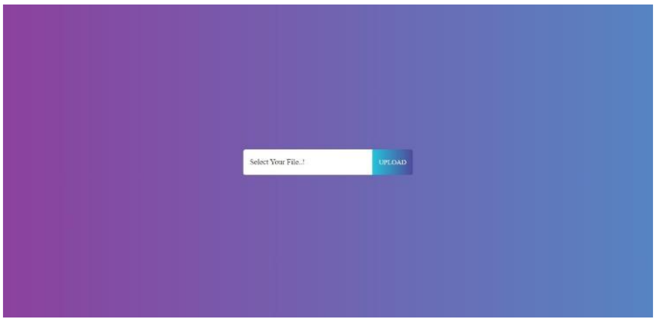


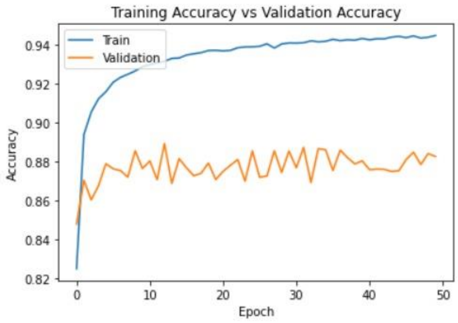
Figure 3. Uploading Answer Scripts

	Keyword-based Summarization	Bag-of-word based Summarization
Precision	0.9	0.83
Recall	0.83	0.41
F-score	0.86	0.53

Figure 4. F-Score Calculation

Human Score	Cosine Approach Score	Error %age
23	33	10
74	72	2
80	72	8
20	34	14
70	95	25
10	17	7
5	0	5
0	9	9
46	34	12
60	79	19

Table 1. Score prediction using cosine similarity.



VI. CONCLUSION

The traditional method of examination entails physical activity such as handling response sheets. Masking is a laborious operation that can take a long time to complete. The goal of this project is to automate the evaluation of subjective answer scripts. It lowers the Examiner's workload in assessing answer scripts This user interface is for converting data.

Handwritten scripts are converted to text scripts, which are then evaluated remotely by the system.The interface is then updated with the evaluated marks. You don't have to keep track of anything as an examiner. Keep note of the compulsory optional questions that students attempt. The system considers For optional responses, take into account the pupils' greatest performances. This system is capable of remove the need to manually enter the marks into the software.

REFERENCES

[1] Piyush Patil and Sachin Patil “Subjective Answer Evaluation using Machine Learning” International Journal for pure and applied Mathematics in 2018. (Vol 118)

[2] Prince Sinha, Sharad, Ayush “Answer Evaluation Using Machine Learning” IEEE 2016

[3]] Amit Rokade, Bhushan Patil, Sana Rajani, Surabhi Revandkar” Automated Grading System usingNatural Language Processing” (IEEE Xplore Compliant 2018 - Part Number: CFP18BAC-ART) .

[4]] M. Syamala Devi and Himani Mittal” MACHINE LEARNING TECHNIQUES WITH ONTOLOGY FOR SUBJECTIVE ANSWER .

[5] 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.

[6] X. Hu and H. Xia, ‘‘Automated assessment system for subjective questions based on LSI,’’ in Proc. 3rd Int. Symp. Intell. Inf. Technol. Secur. Informat., Apr. 2010, pp. 250–254.

[7] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, ‘‘From word embeddings to document distances,’’ in Proc. Int. Conf. Mach. Learn., 2015, pp. 957–966.

[8] C. Xia, T. He, W. Li, Z. Qin, and Z. Zou, ‘‘Similarity analysis of law documents based on Word2vec,’’ in Proc. IEEE 19th Int. Conf. Softw. Qual., Rel. Secur. Companion (QRS-C), Jul. 2019, pp. 354–357.

[9] A. Jabbar, S. Iqbal, M. I. Tamimy, S. Hussain, and A. Akhunzada, ‘‘Empirical evaluation and study of text stemming algorithms,’’ Artif. Intell. Rev., vol. 53, no. 8, pp. 5559–5588, Dec. 2020..

[10] K. Park, J. S. Hong, and W. Kim, ‘‘A methodology combining cosine similarity with classifier for text classification,’’ Appl. Artif. Intell., vol. 34, no. 5, pp. 396–411, Apr. 2020..

[11] S. Aryal, K. M. Ting, T. Washio, and G. Haffari, ‘‘A new simple and effective measure for bag-of-word inter-document similarity measurement,’’ 2019, arXiv:1902.03402.

[12] F. Camastra and G. Razi, ‘‘Italian text categorization with lemmatization and support vector machines,’’ in Neural Approaches to Dynamics of Signal Exchanges(Smart Innovation, Systems and Technologies), vol. 151, A. Esposito, M. Faúndez-Zanuy, F. C. Morabito, and E. Pasero, Eds. New York, NY, USA: Springer, 2020, pp. 47–54.

[13] V. Bahel and A. Thomas, ‘‘Text similarity analysis for evaluation of descriptive answers,’’ 2021, arXiv:2105.02935.

[14] A. Schofield, M. Magnusson, and D. M. Mimno, ‘‘Pulling out the stops: Rethinking stopword removal for topic models,’’ in Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguistics (EACL) vol. 2, M. Lapata, P. Blunsom, and A. Koller, Eds. Valencia, Spain: Association for Computational Linguistics, 2017, pp. 432–436.

[15] M. Çagatayli and E. Çelebi, ‘‘The effect of stemming and stop-wordremoval on automatic text classification in Turkish language,’’ in Proc. 22nd Int. Conf. Neural Inf. Process. (ICONIP) (Lecture Notes in Computer Science), vol. 9489, S. Arik, T. Huang, W. K. Lai, and Q. Liu, Eds. Istanbul, Turkey: Springer, 2015, pp. 168–176

[16] G. Grefenstette, ‘‘Tokenization,’’ in Syntactic Wordclass Tagging. Springer, 1999, pp. 117–133.

[17] 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.