



A
Project Report
on
Automated Copy Checker
submitted as partial fulfillment for the award of
BACHELOR OF TECHNOLOGY
DEGREE

SESSION 2022-23
in
Computer Science & Engineering

By
Ishika Aggarwal (1900290100075)
Pallav Gautam (1900290100095)

Under the supervision of
Prof. Gaurav Parashar
KIET Group of Institutions, Ghaziabad

Affiliated to
Dr. A.P.J. Abdul Kalam Technical University, Lucknow
(Formerly UPTU)
May, 2023

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature 

Name: Ishika Aggarwal, Pallav Gautam

Roll No.: 1900290100075, 1900290100095

Date: 27-May-2023

CERTIFICATE

This is to certify that Project Report entitled “Automated Copy Checker” which is submitted by Ishika Aggarwal and Pallav Gautam in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science & Engineering of Dr. A.P.J. Abdul Kalam Technical University, Lucknow is a record of the candidates own work carried out by them under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

.

Date: 27-May-2023



Prof. Gaurav Parashar

(Assistant Professor)

ACKNOWLEDGEMENT

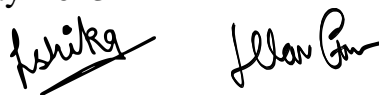
It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe special debt of gratitude to Prof. Gaurav Parashar, Department of Computer Science & Engineering, KIET, Ghaziabad, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.

We also take the opportunity to acknowledge the contribution of Dr. Vineet Sharma, Head of the Department of Computer Science & Engineering, KIET, Ghaziabad, for his full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all the faculty members of the department for their kind assistance and cooperation during the development of our project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members, especially faculty/industry person/any person, of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Date: 27-May-2023

Signature:



Name : Ishika Aggarwal, Pallav Gautam

Roll No.: 1900290100075, 1900290100095

ABSTRACT

A written exam or a test is a technique to check or assess the knowledge of a student or skills and abilities. With every passing year the methods to take examinations change, but what does not change is how those exams are evaluated.

It's the same physical and exhausting mode of evaluation. We used previous research to come up with a solution where the evaluations can be done automatically and with precision and with the least error. We assessed the problems of evaluators and reduced them to almost 90% saving them a huge amount of time.

The main objective of the proposed system is to evaluate the answer sheets with less efforts and with more accuracy. To automate the overall process of evaluating the answer sheets of students, the algorithm proposed comes into play. We converted handwritten notes to text and generated an engine that would take those answer texts and question papers as i/p and then evaluate and give marks. Various concepts, tools and technologies are used in order to accomplish it.

The overall process is divided into five sub-processes. The result we get is on a scale of 0 to 9 then convert them intelligently to percentages. The major part of our system is that we not only focus on the score that is generated by the system, but also provided a human evaluated scores in order to make a comparison between the two obtained result for greater accuracy check and consistency.

A written exam or test is a method for evaluating a student's knowledge, skills, and abilities. Exam taking procedures evolve with each passing year, but how those exams are graded remains constant.

The manner of evaluation is nevertheless physically taxing. In order to create a system where the evaluations can be carried out automatically, precisely, and with the least amount of error, we leveraged prior research. We evaluated the issues facing assessors and approximately 90% of them were eliminated, saving them a significant amount of time.

The major goal of the suggested system is to evaluate the answer sheets more accurately and with less effort. to fully automate the process of assessing the response.

We used previous research to come up with a solution where the evaluations can be done automatically and with precision and with the least error.

Index Terms—Automatic Subjective Evaluation, Intelligent scoring, Automatic feedback, Natural Language Processing, Machine Learning, Similarity Index, Answer Sheets.

TABLE OF CONTENTS

Page No.

DECLARATION.....	ii
CERTIFICATE.....	iii
ACKNOWLEDGEMENTS.....	iv
ABSTRACT.....	v
LIST OF FIGURES.....	ix
LIST OF TABLES.....	x
LIST OF ABBREVIATIONS.....	xi
 CHAPTER 1 (INTRODUCTION).....	 1
1.1. Introduction.....	1
1.2. Project Description.....	3
 CHAPTER 2 (LITERATURE RIVIEW).....	 5
2.1. Summary	5
2.2. Some Examples in this research field	11
2.2.1. Example 1	11
2.2.2. Example 2	26
 CHAPTER 3 (PROPOSED METHODOLOGY)	 31
3.1. Experiment	31
3.1.1. Process I	32
3.1.2. Process II	33
3.1.3. Process III	34
3.1.4. Process IV	35

3.1.5. Process V	36
CHAPTER 4 (RESULTS AND DISCUSSION)	37
CHAPTER 5 (CONCLUSIONS AND FUTURE SCOPE).....	42
5.1. Conclusion.....	42
5.2. Futur Score.....	44
REFERENCES.....	45
APPENDIX	50

LIST OF FIGURES

Figure No.	Description	Page No.
1.	Summarization of actual answers of Students	33
2.	Web Scraping Process	34
3.	Summarization of extracted text	34
4.	Comparison between two GTs	35
5.	Allocation of Marks using Cosine Similarity	36

LIST OF TABLES

Table. No.	Description	Page No.
I	Classes with Ranges	37
II	Cosine Similarity Calculation	39
III	Comparison between Human and Machine Score	40

LIST OF ABBREVIATIONS

ML	Machine Learning
NLP	Natural Language Processing
AI	Artificial Intelligence
i/p	Input
AWE Technologies	Active Wave Engineering Technologies
NLU	Natural Language Understanding
Pdf	Portable Document Format
URL	Uniform Resource Locator
BS4	BeautifulSoup4
GT	Ground Truth

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Educational assessments are very important in the learning process for students. The evaluation and scoring is a tedious process that also takes most of the valuable time of professors. Because of this, the scoring is not optimum, lacks fairness, and is the cause of mental and physical fatigue of the evaluator. Also in physical evaluation, the chances of getting feedback are very difficult. The primary goal of a teacher is, simply, to teach, but this evaluation system is taking away the most important resource a teacher has time.

In this modern age, where the world moves towards automation so, there is a need for automation in answer evaluation system. Currently, the online answer evaluation is available for mcq based question, hence evaluation of the theory answer is hectic for the checker. Teacher manually checks the answer and allot the marks. The current system takes more manpower and time to evaluate the answer.

In this journal an application based on the evaluation of answers using machine learning. The objective of the journal is to specially reduce the manpower and time consumption. Since in manual answer evaluation, the manpower and the time consumption is much more. Also, in the manual system, it may be possible that the marks given to two same answers are different. The theme of the project is to automate the process of evaluation of answer scripts of the student.

In this modern age, there is a need for automation in answer evaluation systems. Currently, online answer evaluation is available for MCQ based questions, but the current system takes more manpower and time to evaluate the answer. This journal proposes an application based on the evaluation of answers using machine learning to reduce the manpower and time

consumption. The objective of the project is to automate the process of evaluation of answer scripts of the student..

Following are some of the major limitations when dealing with subjective answers:

- Existing studies tend to have synonyms.
- Existing studies tend to have an extensive range of possible lengths.
- Existing studies tend to be randomly ordered among their sentences.

So, our system will evaluate answer based on some keyword and also manpower will be saved. Only one has to scan the paper then, based on the keyword in the answer the system will provide the marks to the question according to the dataset present. Also, By this system, the evaluation error of the marks to the particular question will be reduced. So, our system will evaluate answer based on some keyword and also manpower will be saved. Only one has to scan the paper then the system will search the answer on website and provide the URL, based on the keyword in the answer the system will provide the marks to the question according to the dataset present by making comparison using Similarity Index. There is a need for such application which will provide an easy evaluation of answer and can provide eligible marks. Also, this application will help various colleges, university, coaching institute to evaluate the answer in less time and with less manpower.

In this work, we present a system where the answer sheets are evaluated automatically, Automatic Subjective Evaluator. How is an answer sheet actually evaluated? From a teacher's or evaluator's point of view, it's 3 step process. First, you read an answer, then match it with the existing answer we have or judge how close it is to the actual answer, which words are closest to the correct answer, how much is the deviation from "to the point answer" etc., then score according to that. Is a computer program capable of doing all this with precision? Maybe. This task can be achieved using an ability known as Artificial Intelligence (AI). AI is the capability of a computer machine to perform certain tasks without human intervention. AI is the science that learns about human thinking. It is a part of computer science technology that focuses on creating intelligent and quick machines that can perform those tasks that normally require human-like intelligence, such as learning, problem-solving, decision-making, and

language understanding. AI has the power to revolutionize many industries/markets and change the way we live our life and work. It is already helpful in various fields including self-driving cars, virtual assistants, and image processing and speech recognition.

We use natural language processing (NLP) for our system. NLP is a sub-field of artificial intelligence that improves the interaction between computers and humans with the use of natural language. It involves the development of algorithms and models which can analyze, judge, and lead to human language, allowing computers to interpret and respond to human requests and commands. NLP has a vast range of applications, which includes language translation, summarization of text, analysis of sentiments, and development of chatbots. It is a versatile field that joins computer science, linguistics, and cognitive psychology to enable computers to process and understand human language in a way similar to how humans do. We convert those answer sheets into texts then summarise and use them as the input to the engine. After that, it is matched with the answer found on various websites by our engine then uses the similarity index technique to find similarity and score a particular answer.

This application will help various colleges, university, coaching institute to evaluate the answer in less time and with less manpower. This work presents a system where answer sheets are evaluated automatically, called Automatic Subjective Evaluator. AI is the capability of a computer machine to perform certain tasks without human intervention. AI is a part of computer science technology that focuses on creating intelligent and quick machines that can perform tasks that normally require human-like intelligence, such as learning, problem-solving, decision-making, and language understanding.

1.2 PROJECT DESCRIPTION

The main motive of making this system to automate the evaluation process of the answer sheets in order to reduce human efforts. To do this, we have designed a simple algorithm which is divided into five sub-processes and at the end, result is obtained. All these five sub-process have their own working and methodology based on specific techniques.

In our proposed system, we compare the marks given by a machine and a human. Evaluation is done by the algorithm proposed and assign the marks. Then a human will evaluate the same answer and assign the marks. Then a comparison is made between both the marks to find out the accuracy and correctness of the machine evaluation process.

To get the data and use it in our process as input, we gather data from various students by circulating the forms and ask them to fill it. This makes our dataset in order to proceed further with our experiment. After that, our five step sub-processes comes into picture and provide the required output. We get Cosine similarity Index as an output for each answer of the student on a particular question. Then classes are made, based on that scores are assigned.

This system is designed to automate the evaluation process of answer sheets in order to reduce human efforts. It is divided into five sub-processes, each with their own working and methodology based on specific techniques. The system compares the marks given by a machine and a human to find out the accuracy and correctness of the machine evaluation process. The system contains various tools and technology and ML concepts to generate the output. Handwritten notes are converted to text and an engine is generated to take those answer texts and question papers as input and evaluate and give marks.

The data is collected from various students by circulating forms and asking them to fill it. The five step sub-processes provide the required output, including Cosine similarity Index for each answer of the student on a particular question. Classes are made, based on which scores are assigned.

CHAPTER 2

LITERATURE REVIEW

2.1 Summary

An automatic subjective answer sheet evaluator, also known as an automatic essay grader or automated scoring system, is a tool that uses NLP (Natural Language Processing) and algorithms of machine learning to evaluate and grade written responses to open-ended questions. These tools are primarily used in educational settings to grade student essays, but they have also been used in other areas such as language proficiency testing [1].

A literature review of automatic subjective answer sheet evaluators reveals that these tools have been widely researched and developed over the years. Many studies have been conducted to evaluate their effectiveness in grading student essays.

Research has shown that automatic subjective answer sheet evaluators are able to grade written responses with a high level of accuracy. Studies have found that the scores generated by these tools are very close to those produced by human graders.

Past research shows that because of teachers' performance measures (i.e. test scores) damage their job satisfaction, including increased stress and health issues. If we agree with these assumptions, there is an automatic belief that quantitative measures, like test scores, show the reality of the situation and that the result of education is the result of individual actions and is not affected by larger societal contexts or family circumstances. The constant pressure to improve test scores leads to change in the possibilities by which the teaching profession, and teaching professionals, can be valued, and the ways that teachers can ultimately be and associate themselves related to their work.

Many studies have found how the cultural expectations of teachers, and the usual testing culture, are associated with an increase in the workload of the teacher and constant, work related pressure, mental stress, and reduce job satisfaction.

A study has linked risky accountability to excessive fatigue and teacher turnover, arguing that their participants illustrated disappointment, the reality of teaching is worse than we know, and the nature (rather than the quantity) of the workload, linked to performance and accountability, being an important factor for why teachers were leaving the profession.

Bringing together the issues related to statistics by using the test scores of the student in teacher accountability and the creeping pressures often associated with such systems, we debate that the testing culture is bringing an environment where teacher satisfaction is compromised. [1].

Generally, most AWE technologies use (a) Natural Language Processing (NLP) tools to extract linguistic, syntactic, semantic, or different features of text related to writing quality and (b) statistical or machine-learning algorithms to generate scores and feedback based on patterns observed among those features.

Natural language processing(NLP) is a subfield of Artificial Intelligence that deals with the interaction between humans through natural language. NLP enables computers to understand, interpret, and generate human language, whether it is written or spoken. NLP involves various tasks such as text classification, sentiment analysis, named entity recognition, language translation, and speech recognition. It uses machine learning algorithms and statistical models to analyze and derive meaning from human language.

Initially, NLP was not for text retrieval, because it included high-level techniques to search large data sets. As technological advancement grew, that information retrieval and NLP were merged. That is the reason NLP today is so vast. [2]

The goal of NLP is to do human-like processing. Originally the field of NLP was called Natural Language Understanding (NLU). Some of the jobs of NLU were - paraphrasing an input text and translating the text into another language. The concepts that might be used in automatic subjective answer sheet evaluation may be extracting text from an image or a pdf and paraphrasing the questions and answers, drawing inferences from those texts like a human being (the evaluator) does. [3]

Machine learning is the area of study where computers have the ability to learn things without being hard coded. Arthur Samuel was famous for his checkers playing program. ML teaches computers how to handle data correctly. The purpose of ML is to learn from the given data. For that it used algorithms. ML can be supervised or unsupervised. For fewer data, supervised learning is preferred.[4]

An automated answer sheet evaluator helps students get automated scores and feedback, improving their work and writing both in terms of looks and content. It is equally beneficial for the teachers as it gives them the power to monitor and access students more efficiently. Teachers who used AWE gave more effective feedback on answer writing skills than teachers who didn't use it. Due to lack of time and energy, they provide insufficient feedback on marks, answers etc. Still, AWEs are not yet adopted by schools, and universities because of less accurate scoring algorithms. Well, some people argue that if students use automated feedback then their answers will be very much formal and not natural. Some say that NLP-based evaluation will restrict the checking because it will incur only those results that it really can from the methods/algorithms, nothing beyond that. Currently, there is very less amount of studies conducted on AWE.

Questions have also been raised on the conclusions driven by the AWE because it is tested on a certain amount of answers and only in the English language. But it increases the quality of answers produced by the students. It was seen that students who received regular automated feedback produced fewer errors in grammar, style and answer quality. [5]

When tied with tutorials or educational games, learning management features or peer assessment, AWEs potentially offer flexible, robust, and time-saving additions to the writing curriculum.

One of the solutions is to generate each answer again and again for a specific question. It is done using a randomisation array and a key space array. The implementation process is in 3 parts. Generating answers, distributing answers and starting evaluation. Generation starts at the server side by selecting a question. Then the answer is extracted and given to the engine for evaluation. There can be an explanation for this. Giving feedback is different from using

feedback. It was seen that 50% of the students didn't use feedback but those who used improved in a lot of areas.

People should not just consider the cost of buying the AWE software but also what will happen if they will not buy this software. Also with automated feedback student is motivated to write because he/she know that this feedback is unbiased so it increases the effort and outcome of the student. [6]

Now how will this system work? There will be many modules on which the system will work. There may be a management module which has the user information, and login details of both the evaluator and student. The maintainer can put the details of the users and grant permissions. Teachers can maintain their profiles like change passwords, change their subjects, upload study materials, schedule surprise tests etc. The student management portal allows the student to check their result, see feedback both automated and teacher generated, submit solutions, view scores etc.

The answer is scored using the cosine similarity. For every question, there is an answer that the system gets itself from search engines like Google or provided by the teacher. Then both the answers, given by students and by the teacher or the search engine are sent forward to the engine for summarization and other evaluations. Then the similarity between the summarized text is calculated using some algorithms. [7]

AWE is more objective and more accurate while grading an answer. AWS can also score the answer according to the level of the student's mind and his/her needs. It can be programmed for that too. AWE can check a very large number of copies in very less. However, AWE can give unclear judgement and feedback if the student is not familiar with the software's way of evaluating. The way AWE evaluates can be different from what a human evaluates. Also, there will be a fixed number of feedbacks given by the AWE because that feedback will be given from the inbuilt feedback options. The team in this [8] study, worked on the data of 28 undergraduate students. This study showed that there was an increase in the results of the students. The AWE increased their awareness of spelling and grammar. [8]

Most of the research is done by the programmers to make their software work. AWE's feedback may not be useful for everyone or all students. One of the main purposes of AWE is to teach students to go through their answers before submitting their answer sheets which students seriously lack. [9] AWE feedbacks also focus on the revision techniques of the students. It can change the way students revise their copies. [10]

Bahel and Thomas [11] presented an architecture for evaluation of subjective questions using text summarization, text semantics, and keywords summarization and compared the results with existing approaches. The results showed an error of 1.372 compared to 1.312 error from Jaccard's similarity approach. The approach, however, failed to compute nontextual data such as diagrams, images, and other formats.

Jain and Lobiyal [12] proposed a novel approach for subjective questions evaluation using concept graphs. Concept graphs were created for both the solution and the answer, and the score was evaluated using various graph similarity techniques. Montes *et al.* [38] explained various techniques to find Similarities between concept graphs and information retrieval from such graphs.

Wagh and Anand [13] proposed a multi-criteria decision-making perspective to find the Similarity between legal documents. The work included using Artificial Intelligence and aggregation techniques such as ordered weighted average (OWA) for obtaining the similarity value between different documents. Dataset was obtained from Indian Supreme Court case judgments from years ranging from 1950 to 1993. Evaluation measures of F1score and recall were used. As a result, a concept-based similarity approach such as the one proposed in the work performed better than other techniques such as TF-IDF, getting an F1-score of up to 0.8.

Alian and Awajan [14] studied various factors affecting sentence similarity and paraphrasing identification using different word embedding models, clustering algorithms, and weighting methods to find the context of sentences. Pre-trained embeddings included AraVec and FastTex, both trained for the Arabic language. The Arabic training dataset included around 77,600,000 tweets. As a result, pre-trained embedding with labeled data from experts provided better recall and precision of 0.87 and 0.782 for K-means and agglomerative clustering.

Hu and Xia [15] proposed a Latent Semantic Indexing approach for the assessment of subjective questions online. They used Chinese automatic segmentation techniques and subjective ontologies to make a k-dimensional LSI space matrix. The answers were presented in TF-IDF embedding matrices, and then Singular Value Decomposition (SVD) was applied to the term-document matrix, which formed a semantic space of vectors. LSI played the role of reducing problems with synonym and polysemy. At last, the Similarity between answers was calculated using cosine similarity. Dataset consisted of 35 classes and 850 instances marked by teachers, and the results showed a 5% difference in grading done by teacher and the proposed system.

Kusner *et al.* [16] presented a novel concept of using Word Mover's Distance (WMD) to find the dissimilarity between two texts. The system used no hyper-parameters and used a relaxed WMD approach to loosen up the vector space bounds. Dataset included eight real-world sets, including Twitter sentiment data and BBC sports articles. Word2vec model from google news was used, and two other custom models were trained. KNN classification approach was used to classify the testing data. As a result, relaxed WMD reduced the error rates and led to 2 to 5 times faster classification.

We studied various methods used for subjective answer evaluation in the past and looked at their shortcomings. In this paper, we propose a new approach to solve this problem which consists of training a machine learning classification model with the help of results obtained from our result prediction module and then using our trained model to reinforce results from the prediction model, which can lead to a fully trained machine learning model

An automatic subjective answer sheet evaluator, also known as an automatic essay grader or automated scoring system, is a tool that uses NLP and algorithms of machine learning to evaluate and grade written responses to open-ended questions. These tools are primarily used in educational settings to grade student essays, but have also been used in other areas such as language proficiency testing.

A literature review of automatic subjective answer sheet evaluators reveals that these tools have been widely researched and developed over the years. Research has shown that automatic subjective answer sheet evaluators are able to grade written responses with a high level of

accuracy and that the scores generated by these tools are very close to those produced by human graders. Past research has shown that teachers' performance measures (i.e. test scores) damage their job satisfaction, including increased stress and health issues.

The constant pressure to improve test scores leads to change in the possibilities by which the teaching profession, and teaching professionals, can be valued, and the ways that teachers can ultimately be and associate themselves related to their work. A study has linked risky accountability to excessive fatigue and teacher turnover, arguing that their participants illustrated disappointment, the reality of teaching is worse than we know, and the nature (rather than the quantity) of the workload, linked to performance and accountability, being an important factor for why teachers were leaving the profession.

AWE technologies use Natural Language Processing (NLP) tools to extract linguistic, syntactic, semantic, or different features of text related to writing quality, and statistical or machine-learning algorithms to generate scores and feedback based on patterns observed among those features. NLP involves various tasks such as text classification, sentiment analysis, named entity recognition, language translation, and speech recognition. It uses machine learning algorithms and statistical models to analyze and derive meaning from human language. The goal of NLP is to do human-like processing, which is why it is used in automatic subjective answer sheet evaluation.

2.2 Some Examples in the research field

2.2.1 Example 1

ABSTRACT Subjective paper evaluation is a tricky and tiresome task to do by manual labor. Insufficient understanding and acceptance of data are crucial challenges while analyzing subjective papers using Artificial Intelligence (AI). Several attempts have been made to score students' answers using computer science. However, most of the work uses traditional counts or specific words to achieve this task. Furthermore, there is a lack of curated data sets as well. This paper proposes a novel approach that utilizes various machine learning, natural language

processing techniques, and tools such as Wordnet, Word2vec, word mover's distance (WMD), cosine similarity, multinomial naive bayes (MNB), and term frequencyinverse document frequency (TF-IDF) to evaluate descriptive answers automatically. Solution statements and keywords are used to evaluate answers, and a machine learning model is trained to predict the grades of answers. Results show that WMD performs better than cosine similarity overall. With enough training, the machine learning model could be used as a standalone as well. Experimentation produces an accuracy of 88% without the MNB model. The error rate is further reduced by 1.3% using MNB.

INDEX TERMS Subjective Answer Evaluation, Big Data, Machine Learning, Natural Language Processing, Word2vec

INTRODUCTION Subjective questions and answers can assess the performance and ability of a student in an open-ended manner. The answers, naturally, are not bound to any constraint, and students are free to write them according to their mindset and understanding of the concept. With that said, several other vital differences separate subjective answers from their objective counterpart. For one, they are much longer than the objective questions. Secondly, they take more time to write. Moreover, they carry much more context and take a lot of concentration and objectivity from the teacher evaluating them.

Evaluation of such questions using computers is a tricky task, mainly because natural language is ambiguous. Several preprocessing steps must be performed, such as cleaning the data and tokenization before working on it. Then the textual data can be compared using various techniques such as document similarity, latent semantic structures, concept graphs, ontologies. The final score can be evaluated based on Similarity, keywords presence, structure, language. Several attempts have been made in the past to solve this problem, but there is still room for improvements, some of which is discussed in this paper. Subjective exams are considered more complex and scary by both students and teachers due to their one fundamental feature, context.

A subjective answer demands the checker check every word of the answer for scoring actively, and the checker's mental health, fatigue, and objectivity play a massive role in the overall result. Therefore, it is much more time and resource-efficient to let a system handle this

tedious and somewhat critical task of evaluating subjective answers. Evaluating objective answers with machines is very easy and feasible. A program can be fed with questions and one-word answers that can quickly map students' responses. Nevertheless, subjective answers are much more challenging to tackle. They are varied in length and contain a vast amount of vocabulary. Furthermore, people tend to use synonyms and convenient abbreviations, which makes the process that much tricky.

Much work has been done on the topic of subjective answers evaluation in one form or another, 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 the noun-phrase in the documents, matching keywords in the answers, and so on. However, problems such as Tf-Idf losing semantic context, lack of hyper-parameters tuning, costly training, and need for better datasets still exist.

In this paper, we explore a machine learning and natural language processing-based approach for subjective answers evaluation. Our work is based on natural languages processing techniques such as tokenization, lemmatization, text representing techniques such as TF-IDF, Bag of Words, word2vec, similarity measuring techniques such as cosine similarity, and word mover's distance, classification techniques such as multinomial Naive Bayes. We use different evaluation measures such as F1-score, Accuracy, and Recall to evaluate the performance of various models against each other. We also discuss various techniques used in the past for subjective answers evaluation or text similarity evaluation in general.

Following are some of the major limitations when dealing with subjective answers:

- Existing studies tend to have synonyms.
- Existing studies tend to have an extensive range of possible lengths.
- Existing studies tend to be randomly ordered among their sentences.

This paper proposes a new and improved way of evaluating descriptive question answers automatically using machine learning and natural language processing. It uses 2 step approach to solving this problem. First, the answers are evaluated using the solution and provided

keywords using various Similarity-based techniques such as word mover's distance. Then the results from this step are then used to train a model that can evaluate answers without the need for solutions and keywords. For example, a subjective question "What is the capital city of Pakistan and what is it famous for?" can have a correct answer of "Islamabad is the capital city of Pakistan and it is famous for mountain scenery". Before evaluating the student's answer to the question, both the question, the answer, and also some keywords essential to the answer are fed into the system (in this case, keywords will be Islamabad and mountain scenery), and the system evaluates the student's answer by comparing both the similarity (keeping context in mind) of modal answer and student's response as well as the presence or absence of any keywords. So a student's answer of "Karachi is the capital of Pakistan, it is famous for mountain scenery" might get 50% marks, "Islamabad and mountain scenery" might get 30% marks since the main keywords are present even tho context is missing and "Islamabad is the capital and its famous for mountain scenery" might get 100% marks since it satisfies both contextual similarities as well as keywords presence in relation to the correct answer.

A. MOTIVATION This form of evaluation by machines is a big step forward in aiding the educational sector to perform their other duties efficiently and reduce the manual labor in trivial tasks such as comparing the answers with a correct solution in this case. This leads to teachers spending more time teaching students, preparing a better curriculum, and evaluating their tests with less human errors and more transparency.

B. CONTRIBUTION This paper contributes by helping solve the problem of subjective answers evaluation using machine learning and natural processing techniques, it studies various thresholds of sentence similarity measuring matrices and proposes a way to train a machine learning model, which can in turn help reinforce confidence in evaluation score moving forward. Other contributions include a prepared data set with solutions, answers, and keywords carefully curated by teachers.

C. PAPER ORGANIZATION The rest of the paper is organized as follows: Section II presents the background of the problem and the literature review. Section III provides the proposed approach. Section IV presents the experimental analysis and results. Section V concludes the paper.

BACKGROUND AND LITERATURE REVIEW As mentioned before, the evaluation of subjective answers is not a new thought, and it has been worked upon for almost two decades. Various techniques have been implemented to solve this problem, such as big data Natural Language Processing, Latent Semantic Analysis, Bayes theorem, Knearest classifier, and even formal techniques such as Formal Concept Analysis. They are categorized into three main categories: Statistical, Information Extraction, and Full Natural Language Processing.

A. TECHNICAL BACKGROUND

1) Statistical Technique It is based on keyword matching and is considered poor as it cannot tackle problems such as synonyms or take the context into account. Several works have been done on subjective paper evaluation using this approach.

2) Information Extraction (IE) Technique Information Extraction techniques depend on getting a structure or a pattern from the text so that the text can be broken into concepts and their relationships. The dependencies found to play a significant role in producing scores and need to be confirmed from an expert in domain.

3) Full Natural Language Processing(NLP) These techniques involve using natural language tools to parse the text and find its semantic meaning [14], [15]. That meaning can then be compared with the meaning obtained from the solution to assign the final score. Text documents need to be processed and made ready for the machine; this step is called preprocessing and involves various natural language techniques such as Tokenization, Stopword Removal, Parts of Speech Tagging, Lemmatization, Stemming, Case Folding. Some of these techniques are briefly explained below. Nitin et al. [16] discussed automated scoring systems and the use of Natural Language Processing and Machine Learning in them. Zhiwei et al. [17] used Natural Language Processing to measure tree similarity.

4) Tokenization Tokenization is the process of separating data into smaller parts, such as paragraphs, sentences, words, and characters. Tokenization is essential when dealing with natural language because each word must be processed separately to get its true meaning. In this work, we tokenize data into sentences and words based on white spaces and period signs. Tokenization is one of the earliest steps during natural language processing [18]. Kairat et al.

[19] discusses evaluation results of three existing sentence segmentation and word tokenization systems on the Estonian web dataset.

5) Stopword Removal Natural Language has a vast vocabulary, and most elements are there for the ease of human understanding, such as 'the', 'in', 'on', 'is', and so on. These words play little to no role in most machine learning tasks and can even hinder the process by helping the model gets trained on the miscellaneous data. Every language has some known stop words, which are usually removed from the corpus to make the dataset more dense and unique. Alexandra et al. [20] argues that the use of stopword removal is superficial and that topic inference benefits little from the practice of removing stopwords beyond general terms. Mustafa et al. [21] notes that the effect of stopword removal has minimal effect on the actual results as well. However, it should be noted that frequent words with trivial importance for the machine learning model should be removed to improve the model.

6) Parts of Speech Tagging Parts of speech tagging is the processing of tagging each word in the data to its related part of speed, such as a noun, verb, adverb, adjective. Part of speech tagging can be done by various tools such as nltk pos tagger and helps understand the structure of the sentence. It has exciting applications such as finding noun phrases in the sentence, reducing words to their lemma, and so on. Divya et al. [22] used part of speech tagging for efficient sentimental analysis of Twitter.

7) Lemmatization Words found in natural language belong in many forms, such as different tense forms. For example, the words 'go', 'going', 'went' all belong to the same root word 'go' but have different forms. Lemmatization is the process of reducing all the words in the dataset to their root forms. Lemmatization requires a detailed dictionary of the words to relate them to their lemma, also called the root. It also uses part of speed information to relate the words to their specific root in the dictionary. Francesco et al. [23] used Lemmatization and support vector machines to categorize Italian text.

8) Stemming Stemming is a way of reducing words to their stems, and it is based on the idea that every language has some kind of formal grammar, and the vocabulary is formed by always keeping those rules of grammar in mind. So, by using those same rules, we can reduce all the similar words back to their stems by removing their suffixes that make them different.

For example, stemming plurals into singulars (words into words), stemming ending characters, and so on. There are various stemming algorithms forever in every language, such as Porter's algorithm for English word stemming. Jabbar et al. [24] discusses various stemming algorithms used to stem textual data.

9) Case Folding The natural language contains words in different cases, often duplicating the exact words based on their case. Therefore, it is common to reduce all the data into the same case, usually lower case, so that the machine can interpret every word in the same manner. After the preprocessing has been done on the data according to requirements, textual data is converted in a numerical form because machines only understand numbers and understand them very well. This process is called word embedding, and some of the techniques used involve Bag of Words, TFIDF, word2vec.

10) Bag of Words (BoW) Bag of Words is a naive technique that involves representing the vocabulary of the textual data in the form of a vector. That vector contains the index number representing either the count or the particular word at that index in the text. BoW keeps count of frequencies of words but loses the context of words. One example of BoW is a one-hot vector. Sunil et al. [25] used bag-of-words (BoW) vector representation to measure the similarity of two documents concerning each term occurring in the documents.

11) Term Frequency-Inverse Document Frequency (TF-IDF) TF-IDF is similar to BoW, where it counts the frequencies of all words present in the document, but it also keeps track of how many different sentences have those words. This way, it provides information about the count and the value of a word in the document. Sammut et al. [26] discusses Tf-Idf in detail. Havrland et al. [27] gives a probabilistic explanation of TF-IDF approach. Ankit et al. [28] used TF-IDF to predict stock trends.

12) Word2Vec Word2vec is a technique that uses a neural network model to learn word associations from a large dataset. It can be trained for high dimensions such as 300, which helps keep the words' semantic meaning very much intact. After the training is complete, a word2vec model can detect synonymous words or suggest other words based on the sentence. One example of a pre-trained word2vec model is Google News' 300 dimension word2vec model that contains around 100 Billion words.

After the text has been converted into numerical form, aka vectors, it is time to compare those vectors and find the Similarity or dissimilarity between them. Some of the majorly used methods for this task are Cosine Similarity, Jacquard similarity, and Word Mover's Distance. Figure 1 illustrates Word2Vec embedding. Jin et al. [29] studied a semantic similarity computation method based on Word2vec.

13) Cosine Similarity "Cosine similarity is a measure of Similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them. A cosine angle between two vectors is measured, and its value lies between 0 and 1, 1 representing a full match. Park et al. [30] introduced a cosine similarity-based approach to improving the performance of conventional classifiers such as MNB, SVM, and CNN. The cosine of 0° is 1, and it is less than 1 for any other angle in the interval." this method is used extensively in the task of text processing.

14) Jacquard similarity The jacquard similarity is the ratio of intersection to union regarding mutual words. It finds the union of two texts and finds their intersecting terms. Then divide the intersection by its union. The higher the result, the more common words and the bigger the intersection.

15) Word Mover's Distance (WDM) Word Mover's Distance tries to measure the semantic distance of two documents, and word2vec embeddings bring the semantic measurement. Specifically, skip-gram word2vec is utilized in their experiments. Once the word embeddings are obtained, the semantic distance among documents is defined by the following three parts: document representation, similarity metric, and a (sparse)flow matrix. It has been shown to outperform many of the state-of-the-art methods in k nearest neighbors classification [7]. Sato et al. [31] notes that WMD should be superior to BOW because WMD can take the underlying geometry into account, whereas BOW cannot. The similarities obtained from these methods are essentially what we need in order to evaluate a subjective answer.

B. LITERATURE REVIEW Hu et al. [6] proposed a Latent Semantic Indexing approach for the assessment of subjective questions online. They used chinese automatic segmentation techniques and subjective ontologies to make a k-dimensional LSI space matrix. The answers were presented in TF-IDF embedding matrices, and then Singular Value Decomposition

(SVD) was applied to the term-document matrix, which formed a semantic space of vectors. LSI played the role of reducing problems with synonym and polysemy. At last, the Similarity between answers was calculated using cosine similarity. Dataset consisted of 35 classes and 850 instances marked by teachers, and the results showed a 5% difference in grading done by teacher and the proposed system.

Kusner et al. [7] presented a novel concept of using Word Mover's Distance (WMD) to find the dissimilarity between two texts. The system used no hyper-parameters and used a relaxed WMD approach to loosen up the vector space bounds. Dataset included eight real-world sets, including Twitter sentiment data and BBC sports articles. Word2vec model from google news was used, and two other custom models were trained. KNN classification approach was used to classify the testing data. As a result, relaxed WMD reduced the error rates and led to 2 to 5 times faster classification.

Kim et al. [32] proposed a method to grade short descriptive answers lexico-semantic pattern (LSP) due to its good performance with morphologically complex Korean language. LSP can structure the semantic of the answer to help understand the user's intentions. A synonym list was also utilized to help expand the keywords, so they match various answer styles. Dataset was obtained from 88 students and converted to LSP, which was later compared with the solution LSP to score the answer. As a result, the system performed better than the existing system by 0.137

Oghbaie et al. [33] proposed a pair-wise Similarity measure to measure the similarity between two documents based on the keywords which appear in at least one of the documents. The work proposed a new similarity measure called PDSM (pair-wise document similarity measure), a modified version of the preferable properties approach. The proposed similarity measure was applied to text mining applications such as documents detection, k Nearest Neighbors (kNN) for single-label classification, and K-means clustering. An evaluation measure of accuracy was used, and as a result, the PDSM method produced better results than other measures like the Jaccard coefficient by 0.08 recall.

Orkphol et al. [34] used the word2vec approach to represent words on a fix-sized vector space model and then measured the Similarity of sentences using a cosine similarity measure.

Word2vec from google was used, and the sentence vector was obtained as a result of an average of words in the sentence. The score was accepted if it passed a specified threshold for similarity results, between 0 and 1. Evaluation measure of recall and accuracy was used, and as a result, the system's performance was 50.9% with and 48.7% without the probability of sense distribution.

Xia et al. [8] combined the word2vec approach with the legal document corpus to identify similarities between different law documents. Cosine similarity was used to measure the Similarity between different sentence vectors. As a result, word2vec improved the accuracy by 0.2 compared to the Bag of Words approach, which could further be increased by 0.05- 0.10 by training the word2vec model on law documents.

Wagh et al. [35] proposed a multi-criteria decision-making perspective to find the Similarity between legal documents. The work included using Artificial Intelligence and aggregation techniques such as ordered weighted average (OWA) for obtaining the similarity value between different documents. Dataset was obtained from Indian Supreme Court case judgements from years ranging from 1950 to 1993. Evaluation measures of F1score and recall were used. As a result, a concept-based similarity approach such as the one proposed in work performed better than other techniques such as TF-IDF, getting an F1-score of up to 0.8.

Alian et al. [36] studied various factors affecting sentence similarity and paraphrasing identification using different word embedding models, clustering algorithms, and weighting methods to find the context of sentences. Pre-trained embeddings included AraVec and FastTex, both trained for the Arabic language. The Arabic training dataset included around 77,600,000 tweets. As a result, pre-trained embedding with labeled data from experts provided better recall and precision of 0.87 and 0.782 for K-means and agglomerative clustering.

Muangorathub et al. [5] proposed a novel approach of plagiarism detection using formal concept analysis (FCA). The work showed formal context in FCA, starting with two sets containing elements with some attributes that somehow relate the element to its set. The documents and their shared keywords formed a group set in FCA whose values are typically but not limited to 0 and 1. The approach used a many-valued context. The work also

introduced a new similarity concept that uses both the object extent and attribute intent. The approach used is not normally utilized in similarity analysis and ranks similar documents because they have similar object and attribute intents. The proposed system detected plagiarism in documents with 94% accuracy.

Jain et al. [37] proposed a novel approach for subjective questions evaluation using concept graphs. Concept graphs were created for both the solution and the answer, and the score was evaluated using various graph similarity techniques. Montes et al. [38] explained various techniques to find Similarities between concept graphs and information retrieval from such graphs.

Bahel et al. [39] presented an architecture for evaluation of subjective questions using text summarization, text semantics, and keywords summarization and compared the results with existing approaches. The results showed an error of 1.372 compared to 1.312 error from Jaccard's similarity approach. The approach, however, failed to compute nontextual data such as diagrams, images, and other formats.

Table 1 shows the summary of the literature review. We studied various methods used for subjective answer evaluation in the past and looked at their shortcomings. In this paper, we propose a new approach to solve this problem which consists of training a machine learning classification model with the help of results obtained from our result prediction module and then using our trained model to reinforce results from the prediction model, which can lead to a fully trained machine learning model.

The result prediction module uses a word embedding technique called word2vec to generate vectors of our data while keeping their semantic meaning intact and then calculating the Similarity between those vectors using Word Mover's Distance. We compare our results with other methods like using Cosine Similarity and study the effect of using various machine learning models. III.

PROPOSED METHODOLOGY The proposed system consists of data collection and annotation, preprocessing module, similarity measurement module, model training module, results predicting module, machine learning model module, and final result predicting module.

First, the inputs are being taken from the user, which consists of keywords, solutions, and answers. Figure 2 shows the proposed system.

A. KEYWORDS Keywords are question-specific things that are essential for answering that question. These keywords play a significant role in penalizing or promoting the score evaluated by the similarity measurement module and must only contain the essential words in lower case.

B. SOLUTION The solution is a subjective answer that is being used to map students' responses. This solution must contain all the keywords and contexts discussed in the answers in separate lines/paragraphs. The teacher/evaluator typically prepares the solution to the question.

C. ANSWER The answer is a subjective response from the student that is to be evaluated. It usually contains some or all of the keywords and spans 1 to a few sentences depending on the type of question and the student's writing style. It almost always contains synonym words compared to the solution and, therefore, requires much more semantic care when processing.

D. DATA COLLECTION To train and test the proposed model, there is a need for a huge amount of corpus containing subjective question answers, but there is no publicly available labeled subjective question answers corpus to the best of my knowledge. In this work, we create subjective answers labeled corpus. For generating corpus, the important thing is to target those websites and blogs where subjective questions and answers exist. We crawl various websites and collect a subjective question answers corpus. The crawl data belong to various domains such as computer science and general knowledge.

E. DATA ANNOTATION After getting crawled data, there is further need for annotation of data because that crawled data is unlabeled. To annotate data, a group of different volunteers is selected, which belong to the domain of our subjective question answers corpus. We hire 30 different annotators from different colleges and universities and reside in Pakistan's different cities. Most of them are students and teachers. The average age of annotators is in the 21-25 range, whereas some annotators are in the age range of 27-51. We

task annotators to best score the subjective question answers according to the answers given by students.

1) Keyword Generation At the beginning phase of annotation, the data contains just plain answers and no specific keywords. We task annotators to identify the essential terms from the solution which can make or break the overall score of that question. These keywords help decide whether a student has mentioned relevant information in their subjective answers or not.

2) Data Annotation Quality Validation Data validation is crucial to obtaining accurate performance. We perform annotation from three distinct annotators of a single example. We keep the majority voted score as the final annotated label for a particular example.

3) Corpus Statistics Our annotated corpus contained over 1,000 short subjective questions, each containing a correct answer (solution) and 20 students' answers to the question, all of which were annotated. The corpus also contained necessary keywords regarding subjective questions which were extracted from the solutions.

F. PREPROCESSING MODULE After taking inputs from the user, both the solution and the answer go through some preprocessing steps, which involve tokenization, stemming, lemmatization, stop words removal, case folding, finding, and attaching synonyms to the text. Note that stop words are not removed when passing the data to word2vec because word2vec contains a vast vocabulary and can utilize those stop words to make better semantic sense of the text. However, stop words are removed before passing to a machine learning model such as Multinomial Naive Bayes because they hinder the machine's ability to learn the patterns.

G. SIMILARITY MEASUREMENT MODULE This module consists of WDM and Cosine Similarity functions which take two sentences or word vectors and return their Similarity. WDM tells us the dissimilarity while Cosine Similarity measures Similarity. Our approach uses both of these similarity measures one at a time and compares the results at the end. Various similarity (or dissimilarity) thresholds used are given in Table 2.

1) Thresholds Analysis Various thresholds used in this paper have been experimentally deduced to produce the optimal result, WDM thresholds of WDM_LOWER and

WDM_UPPER represent the dissimilarity between two sentences, where more dissimilarity represents high similarity. 0.7 threshold for WDM_LOWER was experimentally observed to represent semantically very similar sentences, and 1.6 thresholds for WDM_UPPER were observed to represent semantically less similar sentences. Anything beyond 1.6 is assumed to be too dissimilar to consider viable for comparison.

Similarly, Cosine similarity thresholds COS_LOWER and COS_UPPER represent the similarity between two sentences, it should be noted that cosine similarity does not take the context of two sentences into account when measuring similarity as opposed to WDM, hence the usage of both of this similarity (or dissimilarity) measuring approaches.

H. RESULT PREDICTING MODULE Result Predicting Module is the core of this work. Figure 3 shows the working of this module. It operates on the following Algorithm 1: We now have the overall score calculated by our module using either WDM or Cosine Similarity while considering the maximum matched solution/answer sentence pairs. This result can be compared to an actual score or fed into a machine learning model to be trained.

I. MACHINE LEARNING MODEL MODULE This model consists of Machine learning models trained on the data obtained from the result prediction module. Its working is as follows:

- Input data from Result Prediction Module.
- Preprocess the solution and answer, removing stop words, and use Countvectorizer to represent them in either Bag of Words or TF-IDF form.
- Convert the overall score obtained from Result Prediction Module into some category. Four categories A, B, C, and D, are used in the paper, representing 1st, 2nd, 3rd, and 4th quarter of a 100. For example, A represents marks from 0 to 25, and B represents 26 to 50.
- The number of categories is kept to a minimum because of the unavailability of the actual dataset. Practically, these categories can be extended to cover smaller score ranges.
- A machine learning model such as Multinomial Naive Bayes, which performs well for multi-class classification, is chosen.
- The preprocessed answer is used as testing data with the machine learning model to predict its class/category, and that category is checked with the result obtained from Result Prediction Module. This gives us confidence in the predicted result from the model.
- The preprocessed answer is fed into the machine learning model along with

its label. Moreover, the model is updated according to new data. • The predicted class is sent to the Final Score Prediction Module along with the solution, answer, and the overall score.

In the beginning, the model is highly sparse and inaccurate, so it is better to train it first with a certain amount of data and then start to test it. The advantage of the model is that it acts as a confidence booster for the Result Prediction Module, provided it has been trained on enough data. Furthermore, it can stand for its own and can be used to predict the grades/class of an answer once it has been trained on enough data. This eliminates the need for word2vec or Result Prediction Module discussed before and produces a model that can be used as a standalone evaluator for that particular question. It also helps deal with the abnormal cases where the Result Prediction Model fails to predict the correct result for a particular answer due to semantic dissimilarity on behalf of the less trained word2vec model.

J. FINAL SCORE PREDICTION MODULE This module is shown in Figure 4; it takes input from the machine learning module and validates the overall score with the class obtained from the machine learning module. Suppose the class matches the score. The score is considered finalized. If the class does not match the score, then the addition or deduction of half the number of values in that range is made based on whether the model suggested score is greater or lesser than the Similarity equivalent score.

It is either assumed that the machine learning model is not trained enough and the score is considered true or if the model has been extensively trained, adjusted score after the model suggestion is considered final, Accepting some inaccuracy from both the Score Prediction the Machine Learning Module.

IV. EXPERIMENTATION AND RESULTS The experiment setup consists of a python notebook running on a web-based Google Colab portal with a RAM of 12 GB and an HDD of 100+ GB. No GPU is turned on for this experiment. A pre-trained word2vec model from Google consisting of 300 dimensions of around 100 Billion words vocabulary is used for this experiment. Corpus was divided into 8:2 ratio representing test and train data, respectively. Train data was used to calculate initial scores from the score prediction modules and train the machine learning model.

Afterward, testing data was fed to the system one by one, updating the machine learning model. The results are obtained using cosine similarity and word mover's distance combined with a Multinomial Naive Bayes model. Both the approaches with and without the model produced results in under a minute at Google Colab. The results are as follows.

CONCLUSION This paper proposed a novel approach to subjective answers evaluation based on machine learning and natural language processing techniques. Two score prediction algorithms are proposed, which produce up to 88% accurate scores. Various similarity and dissimilarity thresholds are studied, and various other measures such as the keyword's presence and percentage mapping of sentences are utilized to overcome the abnormal cases of semantically loose answers. The experimentation results show that, on average word2vec approach performs better than traditional word embedding techniques as it keeps the semantics intact. Furthermore, Word Mover's Distance performs better than Cosine Similarity in most cases and helps train the machine learning model faster. With enough training, the model can stand on its own and predict scores without the need for any semantics checking.

In terms of future improvements, the word2vec model can be trained especially for subjective answers evaluation of a particular domain, and with large data sets, the number of classes or grades in the model can be significantly increased. Subjective answers evaluation remains an interesting problem to tackle, and in the future, we hope to find more efficient ways to solve this problem.

2.2.2 Example 2

ABSTRACT Every year educational institutes conduct various examinations, which include institutional and non-institutional competitive exams. Now a day's online tests and examinations are becoming popular to reduce the burden of the examination evaluation process. The online exams include either objective or multiple-choice questions. Nevertheless, the exams include only objective or multiple-choice questions. However, subjective-based questions and answers are not involved due to the evaluation process complexity and efficiency of the

evaluation process. An automatic answer checker application that checks the written answers and marks the weighted similar to a human being is more helpful in the current modern era is necessary. Hence, the software applications built to check subjective answers may be more useful for allocating marks to the user after verifying the answers for online examination.

INTRODUCTION The Online Examination is beneficial to users as in the present day, and the online exams are based on objective questions and exams are getting digitized all over . In this scenario, exam questions can even be based sub- jective answers.

Meaning that the traditional pen-paper based tests are replaced to computer-based tests that have proven to be :

- (i)more consistent in allocating marks and
- (ii)faster than teachers correcting papers.

The traditional exam usually consisted of subjective answers, which were not the best way of grading the student's perception of the subject. Because sometimes, examiners get bored by checking many answer sheets, and there may be an increase in the false evaluation. Evaluation of such questions using computers is a tricky task, mainly because natural language is ambiguous. Several preprocessing steps must be performed, such as cleaning the data and tokenization before working on it.

BACKGROUND AND LITERATURE REVIEW As mentioned before, the evaluation of subjective answers is not a new thought, and it has been worked upon for almost two decades. Various techniques have been implemented to solve this problem, such as big data Natural Language Processing, Latent Semantic Analysis, Bayes theorem, K-nearest classifiers, and even formal techniques such as Formal Concept Analysis.

They are categorized into three main categories: Statistical, Information Extraction, and Full Natural Language Processing.

A. Technical Background

Statistical Technique: It is based on keyword matching and is considered poor as it cannot tackle problems such as synonyms or take the context into account. Several works have been done on subjective paper evaluation using this approach.

Information Extraction (IE) Technique: Information Extraction techniques depend on getting a structure or a pattern from the text so that the text can be broken into concepts and their relationships. The dependencies found to play a significant role in producing scores and need to be confirmed from an expert in domain.

Literature Survey

1. Online Subjective answer verifying system Using Artificial Intelligence(2021)

Authors: Jagadamba G, Chaya Shree G. Organizations/educational institutes always depend on the grading system through examinations. However, most of the examinations are objective. These systems or any other such system are more advantageous in terms of saving resources but failed to include subjective questions [1, 9, 10]. This paper attempted to evaluate the descriptive answer. The evaluation is done through graphical comparison with a standard answer

2. Subjective Answers Evaluation Using Machine Learning and Natural Language Processing(2021)

Authors: Hamza Arshad, Abdul Rehman Javed. Various methods are used for subjective answer evaluation in the past and looked at their shortcomings. In this paper, we propose a new approach to solve this problem which consists of training a machine learning classification model with the help of results obtained from our result prediction module and then using our trained model to reinforce results from the prediction model, which can lead to a fully trained machine learning model.

3. Tool for Evaluating Subjective Answers using AI(TESA)(2021)

Authors: Shreya Singh, Omkar Manchekar, Ambar Patwardhan All the studies which have been reviewed show that there are various different techniques for the evaluation of subjective answer sheets. The advantage of the system lies in the fact that it uses a weighted average of the closest to accurate techniques to provide the most optimized result. TESA is a systematic and reliable system which eases the role of evaluators and provides faster and more efficient outputs.

4. ASSESS – Automated subjective answer evaluation using Semantic Learning

Authors: Nidhi Dedhia, Kunal Bohra, Prem Chandak This automated approach is beneficial when students need to be assessed online for self improvement. This system gives special emphasis to the specially-abled by providing various speech-based usability features, where the gaps are filled by providing audio facilities like listening to the questions and answering them verbally. The advantage of this system is that it is near completion, has improved performance and caters to a very large audience.

5. Automated Answer-Checker

Authors: Vasu Bansal, M.L. Sharma, Krishna Chandra Tripathi The proposed system could be of great utility to the educators whenever they need to take a quick test for revision purposes, as it saves time and the trouble of evaluating the bundle of papers.

This System would be beneficial for the universities, schools and colleges for academic purpose by providing ease to faculties and the examination evaluation cell.

6. Online Subjective Answer Checker

Authors: Merien Mathew, Ankit Chavan, Siddharth Baikar The project report entitled "Online subjective answer checker" has been developed with much care that it is free of errors and at the same time it is efficient and less time consuming. The important thing is that the system is

robust. Also provision is provided for future developments in the system. The entire system is secured. This online system will be approved and implemented soon.

PROPOSED METHODOLOGY Each answer for coming from a student is evaluated using same pre-processing against answer provided by teacher.

Question Paper : When a teachers logs-in into id, the following three questions are asked:

- a Select from the 2 sets of paper
- b. Select the number of questions
- c. Choose level of complexity

2. Answer sheet

3. Evaluation

Algorithm Navies Bayes Classifier Navie Bayes Classifier is one of the simple and most effective classification algo- rithms which helps in building the fast machine learning models that can make quick predictions.

Natural Language Processing It is the technology is used by machine to understand, analyse, manip- ulate, and interpret human's languages .It helps developers to organize knowledge for performing tasks such as translation, automatic summarization, Named Entity Recognition(NER), relationships extraction, and topics segmentation.

CONCLUSION In this paper, we are design the online subjective answer verifying using artificial intelligence for any sectors like a school, colleges and universities. Hence, the proposed system could be great utility to the educators whenever they need to take a quick test for revision purpose, as it saves time and the trouble of evaluating the bundle of papers.

CHAPTER 3

PROPOSED METHODOLOGY

3.1 Experiment

The system can be used in educational institutions such as schools, colleges, and coaching centres for evaluating the answer sheets of students. In this proposed model, the data set includes typed answers and paragraphs taken from students themselves and from various websites and blogs for comparison. The data set needs to be massive amount so it comes from students through google forms, we also targeted various websites that contain subjective questions answer to train our model. The main theme of our proposed algorithm is to automate the subjective answer sheets evaluation process. Existing system have word limits or they cannot include synonyms in their evaluation process. But In our system, there is no such limitations. Any length of answer which includes similar words can be evaluated and output will be generated.

In our proposed system, we compare the marks given by a machine and a human. Evaluation is done by the algorithm proposed and assign the marks. Then a human will evaluate the same answer and assign the marks. Then a comparison is made between both the marks to find out the accuracy and correctness of the machine evaluation process.

The overall system consists of five sub-processes. All these five sub-process have their own working and methodology based on specific techniques. The idea behind the machine evaluation is that firstly we summarize the subjective question answers written by students and find the first ground truth. Then we search for the actual answer to the subjective question through various websites and platforms in order to find out the second ground truth. A comparison is made and the similarity Index is calculated between both the ground truths obtained using the Cosine Similarity Index.

Cosine similarity is a measure of Similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them. A cosine angle between two vectors is measured, and its value lies between 0 and 1, 1 representing a full match. Park *et*

al. introduced a cosine similarity-based approach to improving the performance of conventional classifiers such as MNB, SVM, and CNN. The cosine of 0° is 1, and it is less than 1 for any other angle in the interval.” this method is used extensively in the task of text processing.

For better results, this algorithm to make comparison is used as it do consider the repetition of words while other algorithms does not. The similarities obtained from these methods are essentially what we need in order to evaluate a subjective answer.

Now, the classes are made based on some defined ranges. Using the value obtained from Cosine similarity, we decide the class and marks are given to the student’s answer.

The machine evaluation process involves summarizing subjective question answers written by students and finding the first ground truth, then searching for the actual answer through various websites and platforms.

Let us understand each process in detail:-

3.1.1 Process I

In this sub-process, the answers written by the students are summarized using the Text summarizer tool QuillBot AI. It is an online AI-based tool that provides the text you need using Natural Language Processing (NLP) by maintaining the context of it. It acts as a Summarizer, Paraphraser, Grammar checker, Plagiarism checker and Citation generator. It is a powerful tool used for research and writing purposes. QuillBot AI is an online AI-based tool that provides the text you need using Natural Language Processing (NLP). After summarization, the first ground truth is obtained, which is the summarized text used for comparison with the student's answer. This GT is the summarized text that is used for comparison with the the student’s answer GT.

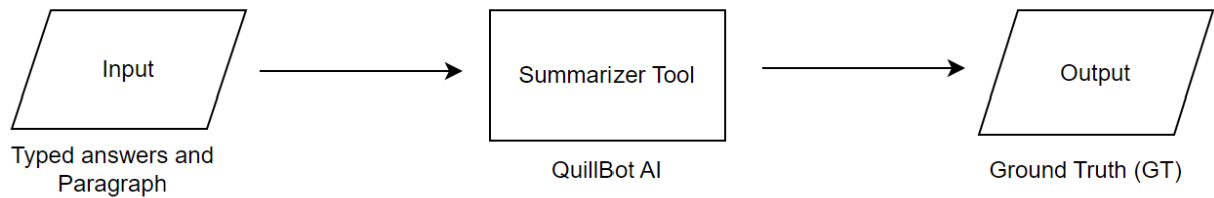


Figure 1. Summarization of actual answers of Students

3.1.2 Process II

In this sub-process, the actual answer to the question is searched over the websites and other platforms and the URL is obtained. The URL is parsed to extract the text from it. The Process is called Web Scrapping which is done using the BeautifulSoup4 package in Python. Web Scrapping is an automated process used to extract the massive amount of unstructured data from websites and store it in a structured format. To do web scrapping in our process, we use Python, its packages and libraries. The output of this process is the extracted text obtained after web scrapping from the particular website.

The steps involve in web scrapping from a particular website provided with the link are as follows:-

- *Step-0* Install python libraries – bs4, requests, html5lib
- *Step-1* Get the HTML
- *Step-2* Parse the HTML
- *Step-3* HTML tree traversal
- Get all the paras from page

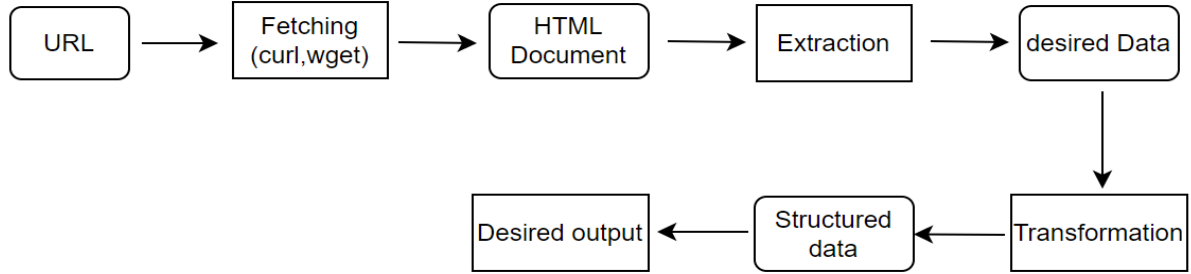


Figure 2. Web Scraping process

3.1.3 Process III

In this sub-process, the parsed text obtained after web scraping is summarized using the summarizing tool QuillBot AI. This will give more precise and brief text. This text is called the second Ground Truth (GT), which is used for comparison with other ground truths. The parsed text obtained after web scraping is summarized using QuillBot AI to give more precise and brief text. This text is called the second Ground Truth (GT) and is used for comparison with other ground truths. Up to this step, both ground truths are obtained and used for comparison with the Cosine Similarity Index algorithm. This process is similar to the Process I, the only difference is the Input that is provided to the system (summarizer tool).

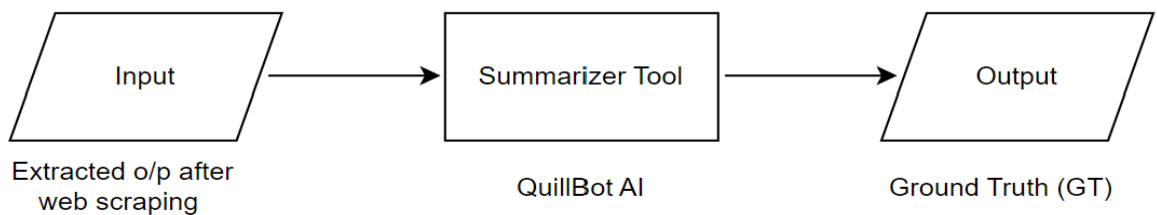


Figure 3. Summarization of Extracted text

3.1.4 Process IV

This sub-process is related to the calculation of the similarity index between two Ground truths(GT) obtained using the Cosine similarity Index. It is implemented using Python. The cosine similarity between two numerical sequences is a metric of similarity. It is defined as the cosine of the angle between the sequences, which is the dot product of the vectors divided by the product of their lengths. The sequences are viewed as vectors in an inner product space. As a result, only the angle of the vectors matters for the cosine similarity, not the magnitudes. The Cosine Similarity Index is a metric of similarity between two numerical sequences. It is defined as the cosine of the angle between the sequences, which is the dot product of the vectors divided by the product of their lengths. The sequences are viewed as vectors in an inner product space, so only the angle of the vectors matters for the cosine similarity. Cosine similarity is always found in the range $[-1,1]$.

The term frequency vectors of the documents are treated as the attribute vectors X and Y for text matching. When comparing documents, cosine similarity may be thought of as a way to normalise document length. It is used to calculate the similarity between two text documents or tokenized texts. The raw data has to be tokenized and a similarity matrix has to be generated which can be passed to the cosine similarity metric to check the similarity between the texts.

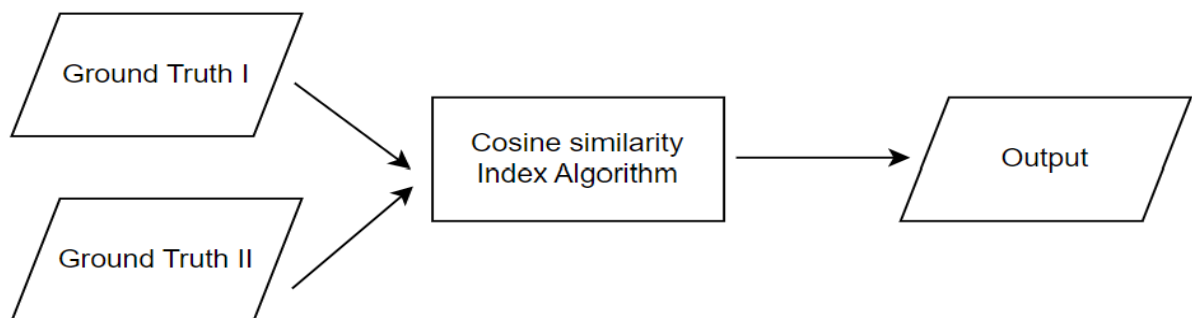


Figure 4. Comparison between two GTs

3.1.5 Process V

The proposed system involves classifying similarity output in ranges to allocate marks to individual answers. When the similarity multiplies by 100, the percentage similarity between two texts is obtained, while when it multiplies by 10, a number ranging from 0 to 9. Classes are made and marks are allotted based on the classes. After the machine evaluation, the same answers are evaluated by a teacher and marks are given. Both marks are compared to find the correctness of the proposed system.

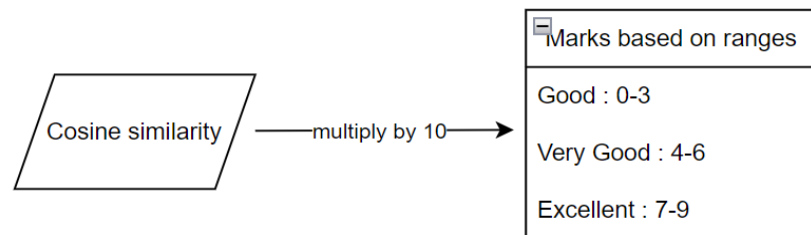


Figure 5. Allocation of marks using Cosine Similarity

CHAPTER 4

RESULTS AND DISCUSSION

a) Division of classes in ranges and allocating Marks

The allocation of marks after calculating the cosine similarity is done by making classes based on the ranges on a scale of 0 to 9. These classes tell us the category of the answer, such as Good, Very Good or Excellent. This helps to make the process of allocation of marks more smooth and organised.

This is shown in table I.

TABLE I
CLASS WITH RANGES

Class	Range
Good	0-3
Very Good	4-6
Excellent	7-9

b) Calculation of Cosine similarity

Using the proposed system, cosine similarity is obtained using the data set from various students. When comparing documents, cosine similarity may be used as a way to normalise the length of the document. The Cosine similarity is calculated between two tokenized texts. Approximate marks or score is given based on the comparison made between the student's answer and the actual answer.

Cosine similarity is always found in the range $[-1,1]$. This number when multiplied by 10 generates the marks scored by the student. This is shown in table II. The proposed system uses cosine similarity to normalize the length of documents. It is calculated between two tokenized

texts and an approximate score is given based on the comparison between the student's answer and the actual answer.

c) Comparison b/w Human Evaluation and Proposed System

In this proposed system, we first show the result obtained by the algorithm developed using the tools and technology. After that, we also evaluated the student's responses manually with the help of teachers in order to check the consistency of the result.

After that, we also evaluated the student's responses manually with the help of teachers in order to check the consistency of the result.

This step plays an important role in checking the accuracy of the proposed system.

This is shown in table III.

TABLE II
COSINE SIMILARITY CALCULATION

Question	Student	Cosine Similarity
Q1	1	0.7035
	2	0.5687
	3	0.4157
	4	0.4747
	5	0.3527
	6	0.5119
	7	0.6080
	8	0.7424
	9	0.7689
Q2	1	0.6424
	2	0.5801
	3	0.6859
	4	0.8891
	5	0.6083
	6	0.6246
	7	0.6080
	8	0.5452
	9	0.9049
Q3	1	0.8314
	2	0.5787
	3	0.7197
	4	0.6624
	5	0.3905
	6	0.7841
	7	0.5746
	8	0.6000
	9	0.9055
Q4	1	0.7428
	2	0.5411
	3	0.6890
	4	0.7105
	5	0.5568
	6	0.6194
	7	0.8528
	8	0.7659
	9	0.5567
Q5	1	0.9028
	2	0.5678
	3	0.6722
	4	0.8567
	5	0.6780
	6	0.5257
	7	0.4896
	8	0.9040
	9	0.6789

TABLE III
COMAPRISON BETWEEN HUMAN AND MACHINE SCORE

Student	Machine Score	Human Score
1	7.0	8.0
2	5.6	8.0
3	4.1	6.0
4	4.7	6.0
5	3.5	6.0
6	5.1	8.0
7	6.0	8.0
8	7.4	8.0
9	7.6	9.0
1	6.4	9.0
2	5.8	8.0
3	6.8	9.0
4	8.8	9.0
5	6.0	9.0
6	6.2	9.0
7	6.0	6.0
8	5.4	6.0
9	9.0	9.0
1	8.3	9.0
2	5.7	6.0
3	7.1	8.0
4	6.6	8.0
5	3.9	4.0
6	7.8	8.0
7	5.7	6.0
8	6.0	8.0
9	9.0	9.0
1	7.4	8.0
2	5.4	6.0
3	6.8	8.0
4	7.1	8.0
5	5.5	6.0
6	6.1	8.0
7	8.5	9.0
8	7.6	8.0
9	5.5	6.0
1	9.0	9.0
2	5.6	6.0
3	6.7	8.0
4	8.5	9.0
5	6.7	9.0
6	5.2	6.0
7	4.8	6.0
8	9.0	9.0
9	6.7	9.0

The current manual evaluation system takes 60 seconds to evaluate an answer, while the proposed system takes 15 seconds. The proposed system is 300% more time efficient and 75-87 accurate than the manual system. It can evaluate 5760 answers in a day, while a human working for 8 hours can evaluate 480 answers a day.

The proposed system eliminates the human effort and time to evaluate an answer, and can evaluate 1100% more answers compared to the manual system. The installation of the proposed system is one time investment with negligible maintenance cost.

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 Conclusion

This paper provides a clear approach to evaluate the answer sheets of the students based on the algorithm proposed using machine learning and natural language processing. Our system provides a convenient way to do evaluations and generate almost accurate results. Various tools and techniques are used in this paper which are better in providing the results. In this, we can evaluate the answer of any length and there are no word limits.

Our system processes and generates the most deserving scores without any partiality. Furthermore, we also make a comparison between the machine score and the human score in order to check the certainty and accuracy of the system. With sufficient training, the model can work on its own and generate scores without the need for any manual semantics checking.

In this paper, we have developed an algorithm which will evaluate theoretical answers and give marks according to the keyword matching which will reduce manual work and saves time with faster result evaluation. A person should collect the answer copy from the student and scan it. The machine will take the image as input and will evaluate the answer based on the length of the answer and important keywords covered which are specified by the teacher with each answer which is to be evaluated.

The algorithm will evaluate theoretical answers and give marks according to the keyword matching, reducing manual work and saving time with faster result evaluation. A person should collect the answer copy from the student and scan it, and the machine will take the image as input and evaluate the answer based on the length of the answer and important keywords covered.

The algorithm assigns marks on basis of :

- The Number of keywords matched.

- Length of the answer

The techniques discussed and implemented in this project should have a high agreement (up to 90 percent) with Human Performance. The project works with the same factors which an actual human being considers while evaluation such as length of the answer, presence of keywords, and context of key-words. Use of Natural Language Processing coupled with robust classification techniques, checks for not only keywords but also the question specific things. Students will have certain degree of freedom while writing the answer as the system checks for the presence of keywords, synonyms, right word context and coverage of all concepts. It is concluded that using ML techniques will give satisfactory results due to holistic evaluation. The accuracy of the evaluation can be increased by feeding it a huge and accurate training dataset. As the technicality of the subject matter changes different classifiers can be employed. Further improvement by taking feedback from all the stakeholders such as students and teachers can improve the system meticulously.

Manual marks are compared with automated scored marks to validate our developed method. In most cases, we have found that our proposed method scored marks similar to the manually assigned marks. It happens for a very few cases that the automated assigned marks are slightly higher or lower than the manually assigned marks. The limitation of our research is that we assign a weight value to each parameter manually by doing a survey. Therefore, our next goal is to introduce machine learning algorithm that will be trained by various calculated parameters, and algorithm will predict the marks of that answer script. Also in the future, we will introduce some new techniques for effective and precise summary generation.

This project uses Natural Language Processing and robust classification techniques to evaluate the length, presence of keywords, and context of key-words. It is concluded that using ML techniques will give satisfactory results due to holistic evaluation. The accuracy of the evaluation can be increased by feeding it a huge and accurate training dataset. Manual marks are compared with automated scored marks to validate the proposed method. The limitation of the research is that we assign a weight value to each parameter manually by doing a survey.

The next goal is to introduce machine learning algorithm that will be trained by various calculated parameters, and algorithm will predict the marks of that answer script. Additionally, new techniques for effective and precise summary generation are also introduced.

5.2 Future Scope

In future, We can extract, handwritten text from the image instead of printed text from image. This will be more realistic and more useful. We can use recursive neural network (RNN) to train our model with different handwriting. This will make our model more accurate.

The model can be trained for different languages across India. In this, we can collect dataset of different handwritten languages. Hence a answer with language other than English can be evaluated. System will also evaluate the overwritten alphabets and other words with absolute accuracy.

The software can be trained in such a way so that, it can check the complete paper instead of a single answer. Hence the software will evaluate the answers according to the answer number provided in the answer sheet. The model can also be trained to evaluate diagrams and hence give the marks accordingly.

The most important details in this text are that a recursive neural network (RNN) can be used to train a model with different handwriting, and that the model can be trained for different languages across India. Additionally, the software can be trained to check the complete paper instead of a single answer, and to evaluate diagrams and give the marks accordingly. This will make the model more accurate and realistic.

REFERENCES

- [1] W. C. Smith and J. Holloway, "School testing culture and teacher satisfaction," *Educational Assessment, Evaluation and Accountability*, vol. 32, no. 4, pp. 461–479, 2020.
- [2] P. M. Nadkarni, L. Ohno-Machado, and W. W. Chapman, "Natural language processing: an introduction," *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 544–551, 2011.
- [3] E. D. Liddy, "Natural language processing," 2001.
- [4] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR)*. [Internet], vol. 9, pp. 381–386, 2020.
- [5] J. Wilson and R. D. Roscoe, "Automated writing evaluation and feedback: Multiple metrics of efficacy," *Journal of Educational Computing Research*, vol. 58, no. 1, pp. 87–125, 2020.
- [6] H. Albazar, "A new automated forms generation algorithm for online assessment," *Journal of Information & Knowledge Management*, vol. 19, no. 01, p. 2040008, 2020.
- [7] T. M. Tashu, J. P. Esclamado, and T. Horvath, "Intelligent on-line exam management and evaluation system," in *Intelligent Tutoring Systems: 15th International Conference, ITS 2019, Kingston, Jamaica, June 3–7, 2019, Proceedings 15*. Springer, 2019, pp. 105–111.
- [8] L. Parra G and X. Calero S, "Automated writing evaluation tools in the improvement of the writing skill." *International Journal of Instruction*, vol. 12, no. 2, pp. 209–226, 2019.
- [9] M. Warschauer and P. Ware, "Automated writing evaluation: Defining the classroom research agenda," *Language teaching research*, vol. 10, no. 2, pp. 157–180, 2006.
- [10] R. D. Roscoe, M. E. Jacovina, L. K. Allen, A. C. Johnson, and D. S. McNamara, "Toward revision-sensitive feedback in automated writing evaluation." in *EDM*, 2016, pp. 628–629.
- [11] M. Islam, *Automated Essay Scoring Using Generalized*, in *Proceedings of 13th International Conference on Computer and Information Technology (ICCIT 2010)*, 2010.

- [12] L. Rudner and T. Liang, Automated essay scoring using Bayes theorem, *J. Technol. Learn.*, vol. 1, no. 2, 2002.
- [13] L. Bin, L. Jun, Y. Jian-Min, and Z. Qiao-Ming, Automated essay scoring using the KNN algorithm, *Proc. - Int. Conf. Comput. Sci. Softw. Eng. CSSE 2008*, vol. 1, pp. 735738, 2008.
- [14] C. Leacock and M. Chodorow, C-rater: Automated scoring of short-answer questions, *Comput. Hum.*, vol. 37, no. 4, pp. 389405, 2003.
- [15] J. Z. Sukkarieh, Using a MaxEnt classifier for the automatic content scoring of free-text responses, *AIP Conf. Proc.*, vol. 1305, pp. 4148, 2010.
- [16] J. Sukkarieh and S. Stoyanchev, Automating Model Building in c-rater, *Proc. 2009 Work.*, no. August, pp. 6169, 2009
- [17] Z. Lin, H. Wang, and S. I. McClean, “Measuring tree similarity for natural language processing based information retrieval,” in *Natural Language Processing and Information Systems, 15th International Conference on Applications of Natural Language to Information Systems, NLDB 2010, Cardiff, UK, June 23-25, 2010. Proceedings* (C. J. Hopfe, Y. Rezgui, E. Métais, A. D. Preece, and H. Li, eds.), vol. 6177 of *Lecture Notes in Computer Science*, pp. 13–23, Springer, 2010.
- [18] G. Grefenstette, “Tokenization,” in *Syntactic Wordclass Tagging*, pp. 117– 133, Springer, 1999.
- [19] K. Sirts and K. Peekman, “Evaluating sentence segmentation and word tokenization systems on estonian web texts,” in *Human Language Technologies - The Baltic Perspective - Proceedings of the Ninth International Conference Baltic HLT 2020, Kaunas, Lithuania, September 22-23, 2020* (U. Andrius, V. Jurgita, K. Jolantai, and K. Danguole, eds.), vol. 328 of *Frontiers in Artificial Intelligence and Applications*, pp. 174– 181, IOS Press, 2020.
- [20] A. Schofield, M. Magnusson, and D. M. Mimno, “Pulling out the stops: Rethinking stopword removal for topic models,” in *Proceedings of the 15th Conference of the European*

Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers (M. Lapata, P. Blunsom, and A. Koller, eds.), pp. 432–436, Association for Computational Linguistics, 2017.

[21] M. Çagataylı and E. Çelebi, “The effect of stemming and stop-word removal on automatic text classification in turkish language,” in Neural Information Processing - 22nd International Conference, ICONIP 2015, Istanbul, Turkey, November 9-12, 2015, Proceedings, Part I (S. Arik, T. Huang, W. K. Lai, and Q. Liu, eds.), vol. 9489 of Lecture Notes in Computer Science, pp. 168–176, Springer, 2015.

[22] M. Divyapushpalakshmi and R. Ramar, “An efficient sentimental analysis using hybrid deep learning and optimization technique for twitter using parts of speech (POS) tagging,” *Int. J. Speech Technol.*, vol. 24, no. 2, pp. 329–339, 2021.

[23] F. Camastra and G. Razi, “Italian text categorization with lemmatization and support vector machines,” in Neural Approaches to Dynamics of Signal Exchanges (A. Esposito, M. Faúndez-Zanuy, F. C. Morabito, and E. Pasero, eds.), vol. 151 of Smart Innovation, Systems and Technologies, pp. 47–54, Springer, 2020.

[24] A. Jabbar, S. Iqbal, M. Ilahi, S. Hussain, and A. Akhunzada, “Empirical evaluation and study of text stemming algorithms,” *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5559–5588, 2020.

[25] S. Aryal, K. M. Ting, T. Washio, and G. Haffari, “A new simple and effective measure for bag-of-word inter-document similarity measurement,” *CoRR*, vol. abs/1902.03402, 2019.

[26] “TF-IDF,” in *Encyclopedia of Machine Learning* (C. Sammut and G. I. Webb, eds.), pp. 986–987, Springer, 2010.

[27] L. Havrlant and V. Kreinovich, “A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation),” *Int. J. Gen. Syst.*, vol. 46, no. 1, pp. 27–36, 2017.

[28] A. Thakkar and K. Chaudhari, “Predicting stock trend using an integrated term frequency-inverse document frequency-based feature weight matrix with neural networks,” *Appl. Soft Comput.*, vol. 96, p. 106684, 2020.

- [29] X. Jin, S. Zhang, and J. Liu, “Word semantic similarity calculation based on word2vec,” in 2018 International Conference on Control, Automation and Information Sciences, ICCAIS 2018, Hangzhou, China, October 24-27, 2018, pp. 12–16, IEEE, 2018.
- [30] 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, 2020.
- [31] R. Sato, M. Yamada, and H. Kashima, “Re-evaluating word mover’s distance,” *CoRR*, vol. abs/2105.14403, 2021.
- [32] J.-E. Kim, K. Park, J.-M. Chae, H.-J. Jang, B.-W. Kim, and S.-Y. Jung, “Automatic scoring system for short descriptive answer written in korean using lexico-semantic pattern,” *Soft Computing*, vol. 22, no. 13, pp. 4241–4249, 2018.
- [33] M. Oghbaie and M. M. Zanjireh, “Pairwise document similarity measure based on present term set,” *Journal of Big Data*, vol. 5, no. 1, pp. 1–23, 2018.
- [34] K. Orkphol and W. Yang, “Word sense disambiguation using cosine similarity collaborates with word2vec and wordnet,” *Future Internet*, vol. 11, no. 5, p. 114, 2019.
- [35] R. S. Wagh and D. Anand, “Legal document similarity: a multi-criteria decision-making perspective,” *PeerJ Computer Science*, vol. 6, p. e262, 2020.
- [36] M. Alian and A. Awajan, “Factors affecting sentence similarity and paraphrasing identification,” *International Journal of Speech Technology*, vol. 23, no. 4, pp. 851–859, 2020.
- [37] G. Jain and D. K. Lobiyal, “Conceptual graphs based approach for subjective answers evaluation,” *Int. J. Concept. Struct. Smart Appl.*, vol. 5, no. 2, pp. 1–21, 2017.
- [38] M. Montes, A. Lopez-Lopez, and A. Gelbukh, “Information retrieval with conceptual graph matching,” vol. 1873, pp. 312–321, 01 2000.
- [39] V. Bahel and A. Thomas, “Text similarity analysis for evaluation of descriptive answers,” *CoRR*, vol. abs/2105.02935, 2021.

[40] A. W. Qurashi, V. Holmes, and A. P. Johnson, “Document processing: Methods for semantic text similarity analysis,” in 2020 International Conference on INnovations in Intelligent SysTems and Applications (INISTA), pp. 1–6, IEEE, 2020.

APPENDIX

Automated Subjective Answer Evaluation Using Machine Learning

1st Ishika Aggarwal

Department of Computer Science & Engineering
KIET Group of Institutions
Ghaziabad, India
ishagrawal24@gmail.com

3rd Gaurav Parashar

Department of Computer Science & Engineering
KIET Group of Institutions
Ghaziabad, India
Orcid Id: 0000-0003-4869-1819

2nd Pallav Gautam

Department of Computer Science & Engineering
KIET Group of Institutions
Ghaziabad, India
pallav.technorex46@gmail.com

Abstract—A written exam or a test is a technique to check or assess the knowledge of a student or skills and abilities. With every passing year the methods to take examinations change, but what does not change is how those exams are evaluated. It's the same physical and exhausting mode of evaluation. We used previous research to come up with a solution where the evaluations can be done automatically and with precision and with the least error. We assessed the problems of evaluators and reduced them to almost 90% saving them a huge amount of time. We converted handwritten notes to text and generated an engine that would take those answer texts and question papers as input and then evaluate and give marks. The result we get is on a scale of 0 to 9 then convert them intelligently to percentages.

Index Terms—Automatic Subjective Evaluation, Intelligent scoring, Automatic feedback

I. INTRODUCTION

Educational assessments are very important in the learning process for students. The evaluation and scoring is a tedious process that also takes most of the valuable time of professors. Because of this, the scoring is not optimum, lacks fairness, and is the cause of mental and physical fatigue of the evaluator. Also in physical evaluation, the chances of getting feedback are very difficult. The primary goal of a teacher is, simply, to teach, but this evaluation system is taking away the most important resource a teacher has time.

In this work, we present a system where the answer sheets are evaluated automatically, Automatic Subjective Evaluator. How is an answer sheet actually evaluated? From a teacher's or evaluator's point of view, it's 3 step process. First, you read an answer, then match it with the existing answer we have or judge how close it is to the actual answer, which words are closest to the correct answer, how much is the deviation from "to the point answer" etc., then score according to that. Is a computer program capable of doing all this with precision? Maybe. This task can be achieved using an ability known as Artificial Intelligence (AI). AI is the capability of a computer machine to perform certain tasks without human in-

tervention. AI is the science that learns about human thinking. It is a part of computer science technology that emphasises on creating intelligent and quick solutions that can execute those tasks that normally require intelligence like humans, for example, learning, solving problems, smart decision making, and language understanding. AI has the power to revolutionize many industries/markets and change the way we live our life and work. It is already helpful in various fields including self-driving cars, virtual assistants, and image processing and speech recognition.

We use natural language processing (NLP) for our system. NLP is a sub-part of artificial intelligence that improves the relationship between computers and humans with the utilization of natural language. It involves the development of algorithms and models which can analyze, judge, and lead to human language, allowing computers to interpret and respond to human requests and commands. NLP has a vast range of applications, which includes language translation, summarization of text, analysis of sentiments, and development of chatbots. It is a flexible field that joins computer science, grammatical, and cognitive psychology to permit computer machines to exercise and understand human language in a familiar way to how we humans do. We convert those answer sheets into texts then summarise and use them as the input to the engine. After that, it is matched with the answer found on various websites by our engine then uses the similarity index technique to find similarity and score a particular answer.

II. LITERATURE REVIEW

An automatic subjective answer sheet evaluator, also known as an automatic essay grader or automated scoring system, is a tool that uses NLP(Natural Language Processing) and algorithms of machine learning to evaluate and grade written responses to open-ended questions. These tools are primarily used in educational settings to grade student essays, but they

have also been used in other areas such as language proficiency testing [1].

A literature review of automatic subjective answer sheet evaluators reveals that these tools have been widely researched and developed over the years. Many studies have been conducted to evaluate their effectiveness in grading student essays.

Research has shown that automatic subjective answer sheet evaluators are able to grade written responses with a high level of accuracy. Studies have found that the scores generated by these tools are very close to those produced by human graders.

Past research shows that because of the performance of teachers' measures, which is the test scores, harms their job fulfilment, together with increased distress and health issues.

If we agree with these kinds of assumptions, there is an automatic trust that numerical measures, like test scores, show the actual reality of the condition and that the result of education is the outcome of the individual working and is not affected by greater societal contexts or family occurrences.

The constant pressure to enhance test scores leads to changes in the probabilities by which the educator profession, and educators, can be valued, and the ways that they can finally ultimately be and relate themselves related to their work.

Many studies have found how the social expectations of educators, and the usual culture of testing, are correlated with a hike in the workload of the teacher and constant, work-related pressure, mental fatigue and stress, and reduced job satisfaction.

A study has linked risky accountability to extravagant fatigue and teacher yield, proclaiming that their participants showed disappointment, the reality of teaching is poorer than we know, and the nature (rather than the quantity) of the workload, associated with performance and accountability, being an important factor for why teachers were leaving the profession.

Bringing together the issues associated with statistics by using the test scores of the student in teacher accountability and the pressures often attached to such systems, we say that the testing culture is bringing an environment where teacher satisfaction is adjusted. [1].

Generally, many of the AWE technologies use (a) Natural Language Processing tools to extract linguistic, syntax-related, semantics-related, or different attributes of text related to writing quality and (b) statistics-related or machine-learning algorithms to create scores and feedback based on patterns discovered among those features.

Natural language processing(NLP) is a subpart of Artificial Intelligence that works with the interaction between humans through natural language. NLP enables computers to understand, explain, and generate human language, whether it is written or spoken. NLP involves various tasks for example classification of text, analysis of sentiments, called entity recognition, language translation, and speech recognition. It uses machine learning algorithms and statistical models to analyze and derive meaning from human language.

Initially, NLP was not for text retrieval, because it included high-level techniques to search large data sets. As technologi-

cal advancement grew, that information retrieval and NLP were merged. That is the reason NLP today is so vast. [2]

The goal of NLP of to do human-like processing. Originally the field of NLP was called Natural Language Understanding (NLU). Some of the jobs of NLU were - paraphrasing an input text and translating the text into another language. The concepts that might be used in automatic subjective answer sheet evaluation may be extracting text from an image or a pdf and paraphrasing the questions and answers, drawing inferences from those texts like a human being (the evaluator) does. [3]

Machine learning is the area of study where computers have the ability to learn things without being hard coded. Arthur Samuel was famous for his checkers playing program. ML teaches computers how to handle data correctly. The purpose of ML is to learn from the given data. For that it used algorithms. ML can be supervised or unsupervised. For fewer data, supervised learning is preferred. [4]

An automated answer sheet evaluator helps students get automated scores and feedback, improving their work and writing both in terms of looks and content. It is equally beneficial for the teachers as it gives them the power to monitor and access students more efficiently. Teachers who used AWE gave more effective feedback on answer writing skills than teachers who didn't use it. Due to lack of time and energy, they provide insufficient feedback on marks, answers etc. Still, AWEs are not yet adopted by schools, and universities because of less accurate scoring algorithms. Well, some people argue that if students use automated feedback then their answers will be very much formal and not natural. Some say that NLP-based evaluation will restrict the checking because it will incur only those results that it really can from the methods/algorithms, nothing beyond that. Currently, there is very less amount of studies conducted on AWE.

Questions have also been raised on the conclusions driven by the AWE because it is tested on a certain amount of answers and only in the English language. But it increases the quality of answers produced by the students. It was seen that students who received regular automated feedback produced fewer errors in grammar, style and answer quality. [5]

When tied with tutorials or educational games, learning management features or peer assessment, AWEs potentially offer flexible, robust, and time-saving additions to the writing curriculum.

One of the solutions is to generate each answer again and again for a specific question. It is done using a randomisation array and a key space array. The implementation process is in 3 parts. Generating answers, distributing answers and starting evaluation. Generation starts at the server side by selecting a question. Then the answer is extracted and given to the engine for evaluation. There can be an explanation for this. Giving feedback is different from using feedback. It was seen that 50% of the students didn't use feedback but those who used improved in a lot of areas.

People should not just consider the cost of buying the AWE software but also what will happen if they will not buy this

software. Also with automated feedback student is motivated to write because he/she know that this feedback is unbiased so it increases the effort and outcome of the student. [6]

Now how will this system work? There will be many modules on which the system will work. There may be a management module which has the user information, and login details of both the evaluator and student. The maintainer can put the details of the users and grant permissions. Teachers can maintain their profiles like change passwords, change their subjects, upload study materials, schedule surprise tests etc. The student management portal allows the student to check their result, see feedback both automated and teacher generated, submit solutions, view scores etc.

The answer is scored using the cosine similarity. For every question, there is an answer that the system gets itself from search engines like Google or provided by the teacher. Then both the answers, given by students and by the teacher or the search engine are sent forward to the engine for summarization and other evaluations. Then the similarity between the summarized text is calculated using some algorithms. [7]

AWE is more objective and more accurate while grading an answer. AWS can also score the answer according to the level of the student's mind and his/her needs. It can be programmed for that too. AWE can check a very large number of copies in very less. However, AWE can give unclear judgement and feedback if the student is not familiar with the software's way of evaluating. The way AWE evaluates can be different from what a human evaluates. Also, there will be a fixed number of feedbacks given by the AWE because that feedback will be given from the inbuilt feedback options. The team in this [8] study, worked on the data of 28 undergraduate students. This study showed that there was an increase in the results of the students. The AWE increased their awareness of spelling and grammar. [8]

Most of the research is done by the programmers to make their software work. AWE's feedback may not be useful for everyone or all students. One of the main purposes of AWE is to teach students to go through their answers before submitting their answer sheets which students seriously lack. [9] AWE feedbacks also focus on the revision techniques of the students. It can change the way students revise their copies. [10]

III. EXPERIMENT

The system can be used in educational institutions such as schools, colleges, and coaching centres for evaluating the answer sheets of students. In this proposed model, the data set includes typed answers and paragraphs taken from students themselves and from various websites and blogs for comparison. The data set needs to be massive amount so it comes from students through google forms, we also targeted various websites that contain subjective questions answer to train our model.

In our proposed system, we compare the marks given by a machine and a human. Evaluation is done by the algorithm proposed and assign the marks. Then a human will evaluate the same answer and assign the marks. Then a comparison is

made between both the marks to find out the accuracy and correctness of the machine evaluation process.

The overall system consists of five sub-processes. All these five sub-process have their own working and methodology based on specific techniques. The idea behind the machine evaluation is that firstly we summarize the subjective question answers written by students and find the first ground truth. Then we search for the actual answer to the subjective question through various websites and platforms in order to find out the second ground truth. A comparison is made and the similarity Index is calculated between both the ground truths obtained using the Cosine Similarity Index. Now, the classes are made based on some defined ranges. Using the value obtained from Cosine similarity, we decide the class and marks are given to the student's answer.

Let us understand each process in detail-

A. Process 1

In this sub-process, the answers written by the students are summarized using the Text summarizer tool QuillBot AI. It is an online AI-based tool that provides the text you need using Natural Language Processing (NLP) by maintaining the context of it. It acts as a Summarizer, Paraphraser, Grammar checker, Plagiarism checker and Citation generator. It is a powerful tool used for research and writing purposes. After summarization, the first ground truth is obtained.

B. Process 2

In this sub-process, the actual answer to the question is searched over the websites and other platforms and the URL is obtained. The URL is parsed to extract the text from it. The Process is called Web Scraping which is done using the BeautifulSoup4 package in Python. Web Scraping is an automated process used to extract the massive amount of unstructured data from websites and store it in a structured format. To do web scraping in our process, we use Python, its packages and libraries.

C. Process 3

In this sub-process, the parsed text obtained after web scraping is summarized using the summarizing tool QuillBot AI. This will give more precise and brief text. This text is called the second ground truth, which is used for comparison with other ground truths.

D. Process 4

This sub-process is related to the calculation of the similarity index between two Ground truths(GT) obtained using the Cosine similarity Index. It is implemented using Python. The cosine similarity between two numerical sequences is a metric of similarity. It is defined as the cosine of the angle between the sequences, which is the dot product of the vectors divided by the product of their lengths. The sequences are viewed as vectors in an inner product space. As a result, only the angle of the vectors matters for the cosine similarity, not the magnitudes. Cosine similarity is always found in the range [-1,1].

The term frequency vectors of the documents are treated as the attribute vectors X and Y for text matching. When comparing documents, cosine similarity may be thought of as a way to normalise document length. It is used to calculate the similarity between two text documents or tokenized texts. The raw data has to be tokenized and a similarity matrix has to be generated which can be passed to the cosine similarity metric to check the similarity between the texts.

E. Process 5

This is the final sub-process which includes classifying similarity output in ranges to allocate marks to individual answers. When similarity multiplies by 100, the percentage similarity between two texts is obtained. When the similarity is multiplied by 10, we get a number ranging from 0 to 9. Classes are made and marks are allotted based on the classes automatically by machine.

After the machine evaluation, the same answers are evaluated by a teacher, and marks are given. Both the marks are compared in order to find the correctness of the proposed system.

IV. RESULTS & DISCUSSION

a) Division of classes in ranges and allocating Marks :

To allocate the marks after calculating the cosine similarity, classes are made based on the ranges on a scale of 0 to 9. These classes tell us the category of answer that whether it is Good, Very Good or Excellent. This will be helpful in finding out the quality of answer based on the category or class, which makes our process of allocation of Marks more smooth and organised. This is shown in table I.

TABLE I
CLASS WITH RANGES

Class	Range
Good	0-3
Very Good	4-6
Excellent	7-9

b) *Calculation of Cosine similarity:* Using the proposed system, cosine similarity is obtained using the data set from various students. When comparing documents, cosine similarity may be used as a way to normalise the length of the document. The Cosine similarity is calculated between two tokenized texts. Approximate marks or score is given based on the comparison made between the student's answer and the actual answer. Cosine similarity is always found in the range $[-1,1]$. This number when multiplied by 10 generates the marks scored by the student. This is shown in table II.

c) *Comparison between Human Evaluation and Proposed System:* In this proposed system, we first show the result obtained by the algorithm developed using the tools and technology. After that, we also evaluated the student's responses manually with the help of teachers in order to check the consistency of the result. This step plays an important role in checking the accuracy of the proposed system. This is shown in table III.

TABLE II
COSINE SIMILARITY CALCULATION

Question	Student	Cosine Similarity
Q1	1	0.7035
	2	0.5687
	3	0.4157
	4	0.4747
	5	0.3527
	6	0.5119
	7	0.6080
	8	0.7424
	9	0.7689
Q2	1	0.6424
	2	0.5801
	3	0.6859
	4	0.8891
	5	0.6083
	6	0.6246
	7	0.6080
	8	0.5452
	9	0.9049
Q3	1	0.8314
	2	0.5787
	3	0.7197
	4	0.6624
	5	0.3905
	6	0.7841
	7	0.5746
	8	0.6000
	9	0.9055
Q4	1	0.7428
	2	0.5411
	3	0.6890
	4	0.7105
	5	0.5568
	6	0.6194
	7	0.8528
	8	0.7659
	9	0.5567
Q5	1	0.9028
	2	0.5678
	3	0.6722
	4	0.8567
	5	0.6780
	6	0.5257
	7	0.4896
	8	0.9040
	9	0.6789

CONCLUSION

This paper provides a clear approach to evaluate the answer sheets of the students based on the algorithm proposed using machine learning and natural language processing. Our system provides a convenient way to do evaluations and generate almost accurate results. Various tools and techniques are used in this paper which are better in providing the results.

In this, we can evaluate the answer of any length and there are no word limits. Our system processes and generates the most deserving scores without any partiality. Furthermore, we also make a comparison between the machine score and the human score in order to check the certainty and accuracy of the system.

With sufficient training, the model can work on its own and generate scores without the need for any manual semantics checking.

TABLE III
COMAPRISON BETWEEN HUMAN AND MACHINE SCORE

Student	Machine Score	Human Score
1	7.0	8.0
2	5.6	8.0
3	4.1	6.0
4	4.7	6.0
5	3.5	6.0
6	5.1	8.0
7	6.0	8.0
8	7.4	8.0
9	7.6	9.0
1	6.4	9.0
2	5.8	8.0
3	6.8	9.0
4	8.8	9.0
5	6.0	9.0
6	6.2	9.0
7	6.0	6.0
8	5.4	6.0
9	9.0	9.0
1	8.3	9.0
2	5.7	6.0
3	7.1	8.0
4	6.6	8.0
5	3.9	4.0
6	7.8	8.0
7	5.7	6.0
8	6.0	8.0
9	9.0	9.0
1	7.4	8.0
2	5.4	6.0
3	6.8	8.0
4	7.1	8.0
5	5.5	6.0
6	6.1	8.0
7	8.5	9.0
8	7.6	8.0
9	5.5	6.0
1	9.0	9.0
2	5.6	6.0
3	6.7	8.0
4	8.5	9.0
5	6.7	9.0
6	5.2	6.0
7	4.8	6.0
8	9.0	9.0
9	6.7	9.0

- [8] L. Parra G and X. Calero S, "Automated writing evaluation tools in the improvement of the writing skill." *International Journal of Instruction*, vol. 12, no. 2, pp. 209–226, 2019.
- [9] M. Warschauer and P. Ware, "Automated writing evaluation: Defining the classroom research agenda," *Language teaching research*, vol. 10, no. 2, pp. 157–180, 2006.
- [10] R. D. Roscoe, M. E. Jacovina, L. K. Allen, A. C. Johnson, and D. S. McNamara, "Toward revision-sensitive feedback in automated writing evaluation." in *EDM*, 2016, pp. 628–629.

REFERENCES

- [1] W. C. Smith and J. Holloway, "School testing culture and teacher satisfaction," *Educational Assessment, Evaluation and Accountability*, vol. 32, no. 4, pp. 461–479, 2020.
- [2] P. M. Nadkarni, L. Ohno-Machado, and W. W. Chapman, "Natural language processing: an introduction," *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 544–551, 2011.
- [3] E. D. Liddy, "Natural language processing," 2001.
- [4] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR)*.*[Internet]*, vol. 9, pp. 381–386, 2020.
- [5] J. Wilson and R. D. Roscoe, "Automated writing evaluation and feedback: Multiple metrics of efficacy," *Journal of Educational Computing Research*, vol. 58, no. 1, pp. 87–125, 2020.
- [6] H. Albazar, "A new automated forms generation algorithm for online assessment," *Journal of Information & Knowledge Management*, vol. 19, no. 01, p. 2040008, 2020.
- [7] T. M. Tashu, J. P. Esclamado, and T. Horvath, "Intelligent on-line exam management and evaluation system," in *Intelligent Tutoring Systems: 15th International Conference, ITS 2019, Kingston, Jamaica, June 3–7, 2019, Proceedings 15*. Springer, 2019, pp. 105–111.