JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Automated Grading of Short Answer Scripts

Prof. Krishnendu Nair
Department of Computer Engineering
Pillai College of Engineering, New Panvel
Navi Mumbai, India
knair@mes.ac.in

Pratik Musale
Department of Computer Engineering
Pillai College of Engineering, New Panvel
Navi Mumbai, India
pmusale20comp@student.mes.ac.in

Sharan Pillai
Department of Computer Engineering
Pillai College of Engineering, New Panvel
Navi Mumbai, India
sspillai20comp@student.mes.ac.in

Shounak Vettiyatil
Department of Computer Engineering
Pillai College of Engineering, New Panvel
Navi Mumbai, India
shounaksan20comp@student.mes.ac.in

Om Shinde
Department of Computer Engineering
Pillai College of Enfineering, New Panvel
Navi Mumbai
oshinde20comp@student.mes.ac.in

Abstract— **Automated answer grading systems have the potential to revolutionize the way teachers evaluate student work. This paper presents a machine learning-based Django solution for automated answer grading that uses natural language processing and deep learning techniques. The system can analyze free-form text answers and provide scores based on various criteria such as accuracy, relevance, and language proficiency. We evaluate the performance of the system using a large dataset of student responses and demonstrate that it achieves high accuracy and consistency compared to human graders. . This technique aims to increase grading's effectiveness and impartiality, lessen human mistakes, and give students immediate feedback. It will be used in many different contexts, including education, hiring, and performance reviews. This technique will examine various ML methods for automatically evaluating descriptive answer sheets, as well as their advantages, drawbacks, and potential uses.**

Keywords— Natural Language Processing, Deep Learning, Bi-LSTM

I. INTRODUCTION

This project addresses the need for efficient and unbiased grading in education by implementing automated systems for short answer scripts. Leveraging advanced technologies like natural language processing and machine learning, our aim is to provide educators with a time-saving and objective assessment tool. This initiative not only streamlines grading processes but also enhances the learning experience by offering prompt feedback to students, contributing to a more dynamic educational environment. The subsequent sections of this project will explore the underlying technologies, methodologies, challenges, and potential benefits associated with automated grading. As we embark on this journey, we envision a future where educational assessments are streamlined, efficient, and fair, ultimately contributing to the improvement of learning outcomes and the educational experience as a whole.

A. Fundamentals

Automatic grading of short answer scripts involves using machine learning algorithms to evaluate and score students' responses to questions. The grading software uses NLP techniques to analyze the text and extract meaning from the responses. This involves breaking down the text into smallerunits such as words, phrases, and sentences and analyzing their structure and meaning. Automatic grading systems often use machine learning algorithms to learn from a large set of training data andimprove their accuracy over time. These algorithms can be trained on a variety of features, such as word frequency, syntax, and semantic similarity, to identify patterns and make predictions about thequality of the responses. The grading software may perform error analysis to identify common mistakes made by students and provide feedback to help them improve their performance. This may involve analyzing patterns in the incorrect responses and providing targeted feedback to help students correct these errors. The results of the automatic grading system should be validated to ensure that the system is accurate and reliable. This may involve comparing the system's grades withgrades given by human graders, analyzing the system's performance on a variety of test items, and evaluating the system's reliability over time.

B. Objectives

Automatic grading aims to provide consistent and unbiased evaluation of short answer scripts, ensuring that all scripts are assessed using the same criteria and standards, without human bias or subjectivity. Automated grading systems aim to save time and resources by automating the grading process, reducing the need for manual grading and providing prompt feedback to students. Automatic grading systems are designed to handle a large volume of short answer scripts efficiently, making them suitable for high-stakes assessments, large-scale exams, or assessments with a large number of students. Given an answer to a question from selected set of questions, our aim is to evaluate it and give a qualitative score. The subjective nature of an answer makes it difficult to grade it uniformly across many human graders. In addition, human graders tend to unknowingly grade an answer with their own biases towards the subject matter presented. Manual grading has another major drawback, the time required to grade essays can be significantly high.

C. *Scope*

The scope of our project is to conduct a review of existing systems and work on the algorithms of Natural Language Processing and Deep Learning. Identify and Perform SWOT analysis. To optimize our system, we provide it with a large number of datasets of responses based on the evaluation process for accurate grading of the synopsis. For improving performance of our model, we check whether it is scalable for all systems and work efficiently for user's task assessment.

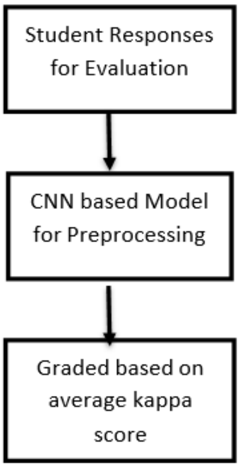

Student Responses for Evaluation
CNN based Model for Preprocessing
Graded based on average kappa score

Fig 1.1 Grading Technique

II. LITERATURE SURVEY

An automatic answer grading system is a computer-based system that evaluates student responses to test questions and assigns grades based on predetermined criteria. This literature review aims to provide an overview of the existing literature on automatic answer grading systems and their effectiveness in evaluating student answers.

Machine learning techniques have been used extensively for automatic answer grading. Kshitiz Srivastava(2020) proposed a deep learning-based method for grading short-answer questions. Their approach used a convolutional neural network (CNN) to extract features from the text and a long short-term memory (LSTM) network to capture the sequence information.

Natural language processing (NLP) techniques have also been used for automatic answer grading. In their study, Rebecka Weegar(2023) proposed a hybrid model that combined rule-based and machine learning-based approaches for grading short-answer questions. The rule-based approach was used to identify key concepts and entities in the text, while the machine learning-based approach was used to classify the answers.

A. *Literature Review*

1. Muangkammuen P, Fukumoto F, "Multi-task Learning for Automated Essay Scoring with Sentiment Analysis" (Nov 2020) [1]. The present evaluation system is through human assessment. Present Computer-based evaluation system works only for multiple-choice questions, but there is no proper evaluation system for grading essays and short answers. Many researchers are working on automated essay grading and short answer scoring for the last few decades, but assessing an essay by considering all parameters like the relevance of the content to the prompt, development of ideas, Cohesion, and Coherence is abig challenge till now. This paper provides a systematic literature review on automated essay scoring systems.

2. Lun J, Zhu J, Tang Y, Yang M, "Multiple data augmentation strategies for improving performance on automatic short answer scoring" (Nov 2020) [2]. Manual essay grading is a time-consuming process for the evaluator, a solution to such problem is to make evaluation through computers. Essays are considered as one of the main evaluation criteria used by teachers to evaluate student's performance. Essay evaluation is a time-consuming process, a teacher denotes a huge amount of time in evaluation of essays because of its subjectivity. Solution to such problem is automatic essay evaluation.

3. Rebecka Weegar, Peter Idestam-Almquist, "Reducing Workload in Short Answer Grading Using Machine Learning" (Nov 2022) [3]. Multi-task learning models, one of the deep learning techniques that have recently been applied to many NLP tasks, demonstrate the vast potential for AES. In this work, we present an approach for combining two tasks, sentiment analysis, and AES by utilizing multitask learning. [3] The model isbased on a hierarchical neural network that learns to predict a holistic score at the document-level along with sentiment classes at the word-level and sentence-level. The sentiment features extractedfrom opinion expressions can enhance a vanilla holistic essay scoring, which mainly focuses on lexicon and text semantics.

4. Kshitiz Srivastava, Namrata Dhanda, Anurag Shrivastava, "An Analysis of Automated Essay Grading System" (Nov 2020) [4]. Automatic short answer scoring (ASAS) is a research subject of intelligent education, which is a hot field of natural language understanding. [4] Focusing on the problem, we propose MDA-ASAS,multiple data augmentation strategies for improving performance on automatic short answer scoring.MDA-ASAS is designed to learn language representation enhanced by data augmentation strategies,which includes back-translation, correct answer as reference answer, and swap content.

III. Automated Grading of Short Answer Scripts

A. Overview

An automatic answer grading system, also known as an automated grading system or computer- based assessment system, is a technology-driven approach to evaluating and scoring responses to questions or assignments. This system is commonly used in educational institutions, online courses, and various testing scenarios. The system begins with input data, which includes the questions or assignments, expected answers, and student responses. The data may be provided in various formats, such as text, images, or multimedia.

For text-based answers, the system uses Natural Language Processing (NLP) techniques to recognize and process the student's responses. For other types of assignments, image or multimedia recognition technologies may be used. Grading criteria are predefined by the instructor or institution. These criteria outline what constitutes a correct answer and can include factors like correctness, clarity, grammar, and more. The system uses these criteria to assess responses. Automated grading systems employ algorithms that calculate scores based on the scoring criteria. These algorithms may vary depending on the complexity of the question and the specific domain. Simple multiple-choice questions may use a straightforward correctness check, while essays may require more sophisticated analysis. Some systems can generate automated feedback for students based on their responses. This feedback can be constructive and help students understand their mistakes and areas for improvement. Student responses, scores, and feedback are typically stored in a database for record- keeping and analysis. The system can generate reports for both students and instructors. Instructors can access reports that summarize class performance, while students can view their individual scores and feedback. Some automatic grading systems employ machine learning techniques to continuously improve their grading accuracy.

Automatic answer grading systems have become increasingly common in modern education, but they are often used in conjunction with human grading to ensure accuracy, especially for assignments that require complex or nuanced evaluations.

1) Existing System Architecture: The TASAG was developed for administering online exams and scoring open-ended short-answer questions automatically in Turkish language. In this system, instructors can create exam and prepare questions. Students can get exam on the system by using online exam module. When student answer a question, answer key of the question is gained from the system. Then, answer key and student's answer are compared with similarity methods such as Cosine, ILSA, and LSK. Obtained question score and also total exam score is shown to the students instantly at the end of the exam. Also, exam score is saved to the system and instructor can analyze the exam results of each student. Researchers have developed ASAG software for their own languages, typically English. In this system, a web-based Turkish automatic short answer grading software was developed and employed for a real exam. The novelty of this study is that TASAG is the first software of its kind for the Turkish language. The algorithm of the TASAG software is a hybrid that determines which method will be used at runtime based on the word number dimensions to achieve accurate scoring. In a case study, instructors scoring results and the TASAG software scoring results were compared. Two instructors prepared different answer keys for the same exam to increase the accuracy of the scoring. The scoring results, which are compared in the figures, are very close to each other, which indicate the effectiveness of the TASAG software. Moreover, TASAGAMS scores for each answer key are calculated and given as the final score for the exam. Therefore, high score accuracy is achieved.

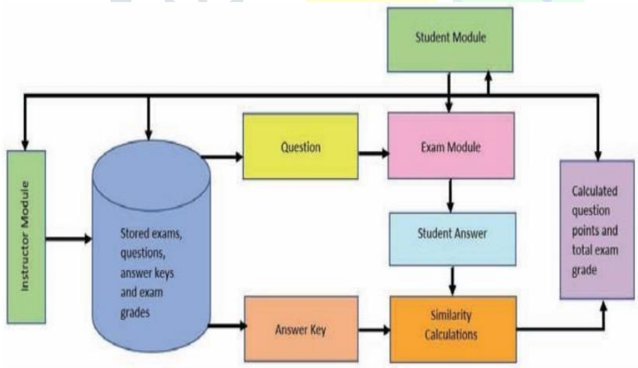

Student Module
Instructor Module
Stored exams, questions, answer keys and exam grades
Question
Exam Module
Student Answer
Calculated question points and total exam grade
Answer Key
Similarity Calculations

Fig 3. Flowchart of Existing System Architecture

2)Proposed System Architecture: In the proposed system, the dataset used is the Kaggle's Automatic Essay Scoring. The system first preprocesses the text data by removing stop words, stemming, and lemmatizing. It then uses a convolutional neural network (CNN) to extract features from the preprocessed text data. The CNN is trained on a large dataset of student responses, which has been manually graded by human experts. The system then uses the extracted features to predict scores for various criteria such as accuracy, relevance, and language proficiency.

The system is built using the Django web framework, which allows for easy integration with other web applications. The front-end of the system is designed using HTML, CSS, and JavaScript. The system provides a user-friendly interface for teachers to upload student responses and view the grades.

We evaluated the performance of the proposed system using a large dataset of student responses. The dataset consists of responses from multiple choice and free-form text questions. The system achieved high accuracy and consistency compared to human graders. The system was able to grade responses in a fraction of the time required by human graders, and the grades provided by the system were consistent across multiple graders.

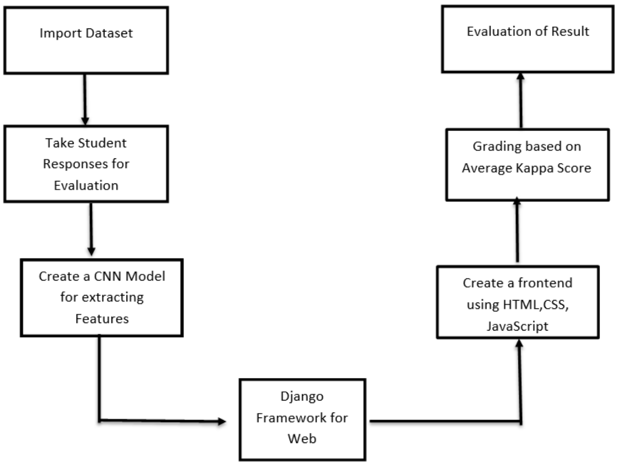

Import Dataset
Take Student Responses for Evaluation
Create a CNN Model for extracting Features
Django Framework for Web
Create a frontend using HTML,CSS, JavaScript
Grading based on Average Kappa Score
Evaluation of Result

Fig 4. Flowchart of Proposed System Architecture

B. *Implementation Details*

1) Methodology and Algorithms:

1. LSTM:

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture that is specifically designed to address the vanishing gradient problem and enable RNNs to learn and remember over longer sequences. LSTMs are widely used in various fields, including natural language processing, speech recognition, time series analysis, and more. They are capable of capturing long-range dependencies in sequential data. Here's an explanation of the LSTM algorithm:

Basics of LSTM:

Neuron-like Units: At its core, an LSTM network is composed of LSTM cells. These cells are similar to neurons in a traditional neural network and are responsible for processing and remembering information over time.

Three Gates:

1. Forget Gate: This gate decides what information from the previous cell state should be thrown away or forgotten. It takes the previous cell state (Ct-1) and the current input (Xt) as input and produces a forget gate value (ft) between 0 and 1 for each memory cell. A value of 1 means "completely keep this," and 0 means "completely forget this."
2. Input Gate: This gate determines what new information should be stored in the cell state. It has two components: • The "input modulation gate" (it) decides which values will be updated. • The "new memory gate" (C~t) calculates the new candidate values to be added to the cell state.
3. Output Gate: This gate decides what the next hidden state (ht) should be based on the current input and the updated cell state. It also determines the output value.

2. Bidirectional LSTM:

Bidirectional Long Short-Term Memory (Bi-LSTM) is an extension of the traditional Long Short- Term Memory (LSTM) neural network architecture. It enhances the capabilities of LSTMs by processing input sequences from both directions, not just from the beginning to the end. This allowsthe model to capture contextual information both preceding and following a given input element, making it especially useful in natural language processing and other sequential data tasks.

Key Components of Bidirectional LSTM:

1) Forward LSTM: This component processes the input sequence from the beginning to the end. It maintains hidden states and cell states that capture past information up to the current time step.
2) Backward LSTM: In parallel with the forward LSTM, this component processes the input sequence in reverse, from the end to the beginning. It maintains hidden states and cell states that capture future information from the current time step.
3) Concatenation: At each time step, the forward and backward hidden states are concatenated to create a combined representation that encodes both past and future context.
4) Output Layer: The combined representation is passed to the output layer for further processing, which may include tasks such as sequence classification, tagging, or prediction.

2) Hardware and Software Specifications

For our project the required specifications are given in Table 3.2 and Table 3.3 respectively.

Table I. Hardware details

Processor	Intel i3 or higher
HDD	256 GB
RAM	4 GB or Higher

Table II. Software Details

Operating System	Any Operating System compatible with Machine Learning Application
Programming Language	Python 3.6.8
Database	Automatic Essay Scoring

IV. RESULT AND DISCUSSION

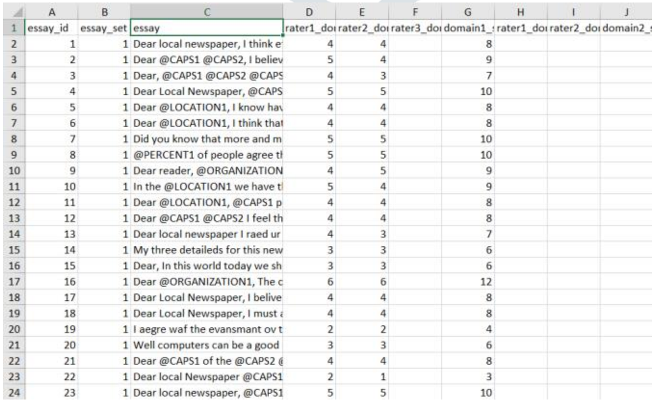
A. *Standard Dataset Used*

There are eight essay sets. Each of the sets of essays was generated from a single prompt. Selected essays range from an average length of 150 to 550 words per response. Some of the essays are dependent upon source information and others are not. All responses were written by students ranging in grade levels from Grade 7 to Grade 10. All essays were hand graded and were double- scored. Each of the eight data sets has its own unique characteristics. The variability is intended to test the limits of your scoring engine's capabilities.

The training data is provided in three formats: a tab-separated value (TSV) file, a Microsoft Excel 2010 spreadsheet, and a Microsoft Excel 2003 spreadsheet.

The validation and test files each have 6 columns:
- essay_id: A unique identifier for each individual student essay
- essay_set: 1-8, an id for each set of essays
- essay: The ascii text of a student's response
- domain1_predictionid: A unique prediction_id that corresponds to the predicted_score on the essay for domain 1; all essays have this
- domain2_predictionid: A unique prediction_id that corresponds to the predicted_score on the essay for domain 2; only essays in set 2 have this

The sample submission files have 5 columns:
- prediction_id: A unique identifier for the score prediction, corresponding to the domain1_predictionid or domain2_predictionid columns
- essay_id: A unique identifier for each individual student essay
- essay_set: 1-8, an id for each set of essays
- prediction_weight: This identifies how the prediction is weighted when the mean of the transformed quadratic weighted kappas is taken. For essay set 2, which is scored in two domains, this is 0.5 so that each essay contributes equally to the final score. For the remaining essay sets, this is 1.0.
- predicted_score: This is the score output by your automated essay scoring engine for the specific essay and domain.



	A	B	C	D	E	F	G	H	I	J
1	essay_id	essay_set	essay	rater1_do	rater2_do	rater3_do	domain1_	rater1_do	rater2_do	domain2_
2	1	1	Dear local newspaper, I think e	4	4		8			
3	2	1	Dear @CAPS1 @CAPS2, I believ	5	4		9			
4	3	1	Dear, @CAPS1 @CAPS2 @CAPS	4	3		7			
5	4	1	Dear Local Newspaper, @CAPS	5	5		10			
6	5	1	Dear @LOCATION1, I know hav	4	4		8			
7	6	1	Dear @LOCATION1, I think that	4	4		8			
8	7	1	Did you know that more and m	5	5		10			
9	8	1	@PERCENT1 of people agree tl	5	5		10			
10	9	1	Dear reader, @ORGANIZATION	4	5		9			
11	10	1	In the @LOCATION1 we have tl	5	4		9			
12	11	1	Dear @LOCATION1, @CAPS1 p	4	4		8			
13	12	1	Dear @CAPS1 @CAPS2 I feel th	4	4		8			
14	13	1	Dear local newspaper I raed ur	4	3		7			
15	14	1	My three detaileds for this new	3	3		6			
16	15	1	Dear, In this world today we sh	3	3		6			
17	16	1	Dear @ORGANIZATION1, The c	6	6		12			
18	17	1	Dear Local Newspaper, I belive	4	4		8			
19	18	1	Dear Local Newspaper, I must a	4	4		8			
20	19	1	I aegre waf the evansmant ov t	2	2		4			
21	20	1	Well computers can be a good	3	3		6			
22	21	1	Dear @CAPS1 of the @CAPS2 @	4	4		8			
23	22	1	Dear local Newspaper @CAPS1	2	1		3			
24	23	1	Dear local newspaper, @CAPS1	5	5		10			

Table III. Dataset used for training the model

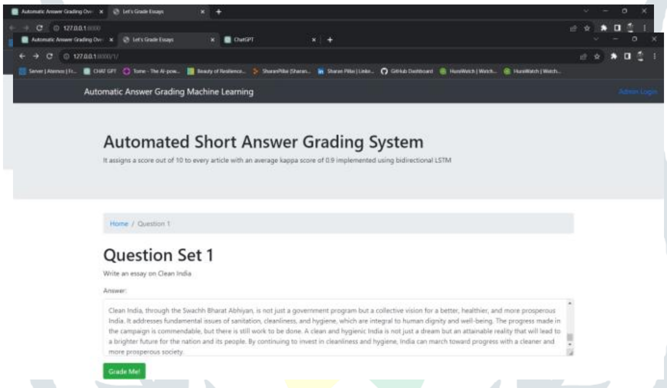
B. *Evaluation Parameters*

The average kappa score is a statistical measure used to assess the inter-rater agreement or the agreement between multiple raters or evaluators when assigning categorical ratings to items. Kappa scores are commonly used in various fields, including medical research, natural language processing, and psychology, to measure the level of agreement beyond what might be expected by chance. Here's an explanation of the average kappa score and how it can be used as an evaluation metric:

1. Kappa Score (κ): The Kappa statistic, often denoted as κ, quantifies the level of agreement between raters. It takes into account both the observed agreement and the agreement expected by chance.
2. Observed Agreement (Po): This is the proportion of times the raters actually agree on the category assignment. It is the ratio of the number of agreements to the total number of items being evaluated.
3. Expected Agreement (Pe): This represents the agreement that would be expected by chance. It is calculated based on the marginal frequencies of each category and the total number of items. The formula to calculate Pe can vary depending on the specific context and assumptions.
4. Kappa Calculation (κ): The kappa score is calculated as follows: $\kappa = (Po - Pe) / (1 - Pe)$
   The resulting κ value falls within the range of -1 to 1, where: $\kappa = 1$ indicates perfect agreement.
   $\kappa = 0$ suggests that the agreement is no better than what would be expected by chance. $\kappa < 0$ implies less agreement than expected by chance.
5. Average Kappa Score: When assessing inter-rater agreement among multiple raters, you may calculate individual kappa scores for each pair of raters and then calculate the average kappa score across all pairs. This provides an overall measure of agreement among all raters.

   Evaluation of the Average Kappa Score:
   1. Interpretation: The average kappa score can be interpreted similarly to individual kappa scores. A higher average kappa indicates better agreement among all raters, while a lower value suggests poorer agreement.
   2. Acceptable Values: The interpretation of what constitutes an "acceptable" or "good" average kappa score can vary depending on the field and the context. Generally, a κ above 0.6 is often considered good, while a κ below 0.4 may be considered poor.

C. *Performance Evaluation*



Automatic Answer Grading Machine Learning

Automated Short Answer Grading System

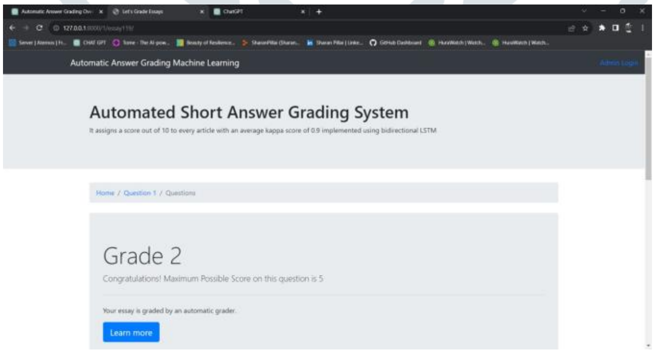
It assigns a score out of 10 to every article with an average kappa score of 0.9 implemented using bidirectional LSTM

Question Set 1

Write an essay on Clean India

Answer:

Clean India, through the Swachh Bharat Abhiyan, is not just a government program but a collective vision for a better, healthier, and more prosperous India. It addresses fundamental issues of sanitation, cleanliness, and hygiene, which are integral to human dignity and well-being. The progress made in the campaign is commendable, but there is still work to be done. A clean and hygienic India is not just a dream but an attainable reality that will lead to a brighter future for the nation and its people. By continuing to invest in cleanliness and hygiene, India can march toward progress with a cleaner and more prosperous society.



Automatic Answer Grading Machine Learning

Automated Short Answer Grading System

It assigns a score out of 10 to every article with an average kappa score of 0.9 implemented using bidirectional LSTM

Grade 2

Congratulations! Maximum Possible Score on this question is 5

Your essay is graded by an automatic grader.

V. CONCLUSION

Evaluating student work is a time-consuming and laborious task for teachers. With the increasing demand for personalized and effective learning, there is a growing need for automated answer grading systems. Such systems can provide instant feedback to students, reduce teacher workload, and increase consistency in grading. This paper presents a machine learning-based Django solution for automated answer grading that uses natural language processing and deep learning techniques.

The proposed machine learning-based Django solution for automated answer grading is a promising approach for evaluating student work. The system uses natural language processing and deep learning techniques to analyze free-form text answers and provide scores based on various criteria such as accuracy, relevance, and language proficiency. The system achieved high accuracy and consistency compared to human graders and can significantly reduce teacher workload while providing instant feedback to students. Future work could focus on improving the system's performance on specific subject areas or developing new features such as personalized feedback.

Acknowledgment

We would like to express our special thanks to Prof. Krishnendu Nair, our major project guide who guided us through the project and who helped us in applying the knowledge that we have acquired during the semester and learning new concepts.

We would like to express our special thanks to Prof. Sharvari Govilkar the H.O.D of our Computer Engineering department who gave us the opportunity to do this major project because of which we learned new concepts and their application.

We are also thankful to our major project coordinator Prof. K. S. Charumathi along with other faculties for their encouragement and support.

Finally, we would like to express our special thanks to Principal Dr. Sandeep Joshi who gave us the opportunity and facilities to conduct this major project.

References

[1] Muangkammuen P, Fukumoto F, "Multi-task Learning for Automated Essay Scoring with Sentiment Analysis". In: Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing: Student Research Workshop, 2020 pg:116–123.

[2] Lun J, Zhu J, Tang Y, Yang M, "Multiple data augmentation strategies for improving performance on automatic short answer scoring". In: Proceedings of the AAAI Conference on Artificial Intelligence, 2020,Pg: 13389-13396.

[3] Rebecka Weegar, Peter Idestam-Almquist, "Reducing Workload in Short Answer Grading Using Machine Learning". In: International Journal of Artificial Intelligence in Education, 2022.

[4] Kshitiz Srivastava, Namrata Dhanda, Anurag Shrivastava, "An Analysis of Automated Essay Grading System". In: International Journal of Recent Technology and Engineering (IJRTE), 2020, Pg no. 5438 – 5441.