Karbala International Journal of Modern Science

Manuscript 3370

IntelliGrader: A Framework for Automatic Short Answer Grading, Inconsistency Check and Feedback in Educational Context - Conception, Implementation and Evaluation

Paradesi Sree Lakshmi

Jay B. Simha

Rajeev Ranjan

IntelliGrader: A Framework for Automatic Short Answer Grading, Inconsistency Check and Feedback in Educational Context - Conception, Implementation and Evaluation

Abstract

Automatic Short Answer Grading (ASAG), an escalating realm in natural language understanding, constitutes a focal point of research within the broader field of learning analytics. Over time, many ASAG solutions have been proposed to address the difficulties in teaching. However, no work addressed three crucial aspects of evaluation together, i.e., i) automatic evaluation of brief subjective/descriptive answers written in English, ii) identifying the evaluation inconsistency, and iii) provision of providing feedback about inconsistent evaluation to the evaluator. The current work proposes IntelliGrader, a comprehensive ASAG system that addresses the above-mentioned issues. Automated grading is accomplished through a model answer-based approach. The collaborative analysis of eight crucial features, incorporating statistical, word-word, keyword, lemmatized, term frequency-inverse document frequency, contextual, semantic, and summary resemblances amid model and student answers, are performed, utilizing state-of-the-art regressors. In contrast to other existing research, i) Unsupervised learning approaches were used to identify inconsistencies in evaluation, ii) Underwent rigorous validation on benchmark datasets ASAP, STITA, and a novel dataset IDEAS. Experimental results show the finest Root Mean Square Error of 0.09 on the STITA dataset and 0.19 on IDEAS for a specific question. IntelliGrader performs better than the systems presented in the literature. Experimental results regarding the inconsistency showed less inconsistency in model-predicted scores when compared with human evaluation, showing the model's accuracy. Finally, the identified inconsistency is provided as detailed feedback to the evaluator, which assists them in improving the evaluation process. We recommend using this as a tool to support evaluators, not to replace human judgment.

Keywords

Automatic Short Answer Grading (ASAG); Regression; clustering; K Means; Inconsistency; Feedback

Creative Commons License

RESEARCH PAPER

IntelliGrader: A Framework for Automatic Short Answer Grading, Inconsistency Check and Feedback in Educational Context-conception, Implementation and Evaluation

Paradesi S. Lakshmi [a,*], Jay B. Simha [b], Rajeev Ranjan [a]

[a] School of Computer Science and Applications, REVA University, Bengaluru, India
[b] REVA Academy for Corporate Excellence (RACE), REVA University, India

Abstract

Automatic Short Answer Grading (ASAG), an escalating realm in natural language understanding, constitutes a focal point of research within the broader field of learning analytics. Over time, many ASAG solutions have been proposed to address the difficulties in teaching. However, no work addressed three crucial aspects of evaluation together, i.e., i) automatic evaluation of brief subjective/descriptive answers written in English, ii) identifying the evaluation inconsistency, and iii) provision of providing feedback about inconsistent evaluation to the evaluator. The current work proposes IntelliGrader, a comprehensive ASAG system that addresses the above-mentioned issues. Automated grading is accomplished through a model answer-based approach. The collaborative analysis of eight crucial features, incorporating statistical, word-word, keyword, lemmatized, term frequency-inverse document frequency, contextual, semantic, and summary resemblances amid model and student answers, are performed, utilizing state-of-the-art regressors. In contrast to other existing research, i) Unsupervised learning approaches were used to identify inconsistencies in evaluation, ii) Underwent rigorous validation on benchmark datasets ASAP, STITA, and a novel dataset IDEAS. Experimental results show the finest Root Mean Square Error of 0.09 on the STITA dataset and 0.19 on IDEAS for a specific question. IntelliGrader performs better than the systems presented in the literature. Experimental results regarding the inconsistency showed less inconsistency in model-predicted scores when compared with human evaluation, showing the model's accuracy. Finally, the identified inconsistency is provided as detailed feedback to the evaluator, which assists them in improving the evaluation process. We recommend using this as a tool to support evaluators, not to replace human judgment.

Keywords: Automatic Short Answer Grading (ASAG,), Regression, Clustering, K Means, Inconsistency, Feedback

1. Introduction

Evaluation is a crucial aspect of education. Answer grading serves as a process to allocate a numerical value that implies the quality or the level of student performance. The integrity of the grading process is vital in education. Nevertheless, achieving a fair and consistent assessment and timely feedback remains challenging. Automatic short answer grading (ASAG) or Computer-Assisted Assessment (CAA) is a reliable substitute for the manual evaluation process [1]. Academic examinations may employ diverse question formats. From the grading standpoint, questions are generally classified into two main categories: objective questions (Ex. fill-in-the-blank, true or false, Multiple Choice) and subjective questions, which involve open-ended (essays) and close-ended (short answers) formats [2]. Concerning objective questions, students must choose one among the given options. In contrast, in subjective, they must write the answers in their own words by going beyond simple recall, which prompts

Received 10 April 2024; revised 2 September 2024; accepted 3 September 2024.
Available online 9 October 2024

[*] Corresponding author.
E-mail addresses: p.sreelakshmi@reva.edu.in (P.S. Lakshmi), jb.simha@reva.edu.in (J.B. Simha), rajeev.ranjan@reva.edu.in (R. Ranjan).

https://doi.org/10.33640/2405-609X.3370

higher-order student thinking [3]. Nevertheless, grading constructed response items demands considerable time and effort, and the resulting grades' consistency is frequently a concern [4].

In this paper, three crucial aspects of evaluation are addressed. i) Automatic evaluation of brief descriptive answers, ii) Identification of evaluation inconsistency, and iii) provide comprehensive feedback about inconsistently evaluated responses to the assessor/evaluator. Now, we discuss each aspect's importance and the corresponding literature limitations.

Regarding the first issue, i.e., grading short descriptive answers automatically, from the existing literature, the fundamental machine-learning approaches for ASAG systems are categorized into two groups [5]. The first approach utilizes a classification method to assign scores logically, categorizing responses as correct, incorrect, partially correct, or contradictory [5−9]. The primary limitation of this method is its inability to assign numerical scores to answers; instead, it categorizes them simply as either correct or incorrect. The second approach is built on regression, assigning numerical scores like 4.3, 2.5, etc. [10−13]. The primary disadvantage of these approaches is that they require a learning process, such as rounding to the nearest integer, to convert real scores into integer scores, given that they generate decimal values.

Concerning the second issue, inconsistency identification, Intra-rater inconsistency arises from mood fluctuations, personal bias, or contextual influences. It disrupts expected evaluation standards by leading evaluators to assess similar answers differently on separate occasions. The literature regarding the identification of inconsistency in the evaluation is scarce [14,15]. Inconsistency in evaluation compromises reliability, leading to inaccurate or biased results. Identifying and addressing this inconsistency is crucial for maintaining the assessment's integrity, reliability, and fairness, and it helps identify areas needing improvement.

The third aspect, i.e., timely feedback about inconsistent evaluations, aids in identifying and correcting grading errors, thereby promoting equity and fairness within the assessment system. This proactive approach enhances quality assurance by pinpointing recurrent grading inconsistencies and mitigating subjective biases. In the realm of literature, limited efforts have been made to give feedback to the students regarding their responses [16−19]. To our knowledge, no systems provide feedback on inconsistent answers to the evaluator.

1.1. The overarching objective of this work

The core focus of the current work is on developing an ASAG framework, *IntelliGrader*, that supports:

- Evaluators/instructors in assessing short descriptive answers in English.
- Detecting inconsistencies in assessment.
- Provides feedback to the evaluator concerning inconsistently evaluated responses, enhancing the assessment process's fairness, quality, and effectiveness, benefiting both students and the educational institution.

On this page, our objective is to respond to the subsequent research queries:

i) What is the optimal strategy for ASAG in the setting of *IntelliGrader*? This investigation examines the features or text characteristics that should be considered and determines the most appropriate machine-learning method for effective ASAG.

ii) How does the performance of the suggested approach vary across diverse datasets?

1.2. Proposed approach

In this context, a new framework called *IntelliGrader* is introduced, which assists evaluators in three essential evaluation tasks. It serves as a screening tool for automatically grading short descriptive answers, identifying inconsistencies, and offering detailed feedback on inconsistently evaluated responses/answers to the evaluator. Here, an answer-based model approach is used. Eight similarities between the student and model answers are exploited as features to provide a final score. The proposed method combines the traditional NLP tasks, including Bag of Words, along with state-of-the-art deep learning techniques involving Infersent [20] sentence embeddings for semantic analysis. Section 3.4 details the need and importance of considering these eight features. Regression models are designed using sophisticated regressors, with these eight characteristics as independent variables and the marks assigned by the evaluator as the dependent variable. Additionally, an unsupervised learning approach, K Means, identifies inconsistencies in student answers, with a mechanism providing feedback to evaluators.

1.3. Methodological contributions

The key findings of the present study are abridged as follows:

- Introduced *IntelliGrader*, an ASAG framework that performs three significant tasks, including automatic grading of brief descriptive responses that are written in English, performing inconsistency checks, and feedback regarding the inconsistent answers to the evaluator.
- *IntelliGrader* is innovative since it simultaneously addresses eight features of short answers encompassing all aspects of comparing the student and the model answer.
- Unlike existing literature, comprehensive experiments are conducted question-wise across publicly available datasets ASAP, STITA, and new dataset *IDEAS.* The experimental results indicate the proposed methods perform on par with and at times exceed the latest approaches.
- A novel dataset, *IDEAS*, has been unveiled for ASAG, which will serve as the testing ground for our proposed solution. The dataset will be publicly available shortly, catering to all researchers interested in this domain.
- In contrast to the prevailing literature, a novel method is proposed for identifying inconsistencies in student answer evaluation. The feedback report regarding inconsistent answers is immediately provided to the evaluator, which enhances the evaluation process.

The paper is systematized as follows: Initially, Section 2 provides an overview of relevant literature and includes a schematic comparison between those works and the one presented in this study. Section 3 presents the proposed methodology, while Section 4 shows the achieved results and is thoroughly discussed to provide insights. Finally, Section 5 concludes with final remarks and outlines future developments.

2. Literature review

This section provides details and reports on previous literature, highlighting points of contact with the work presented here. In this section, we delve into the literature covering three crucial evaluation tasks: ASAG, inconsistency check, and feedback to the evaluator.

First, concerning the literature in ASAG, in the study [2] they explored various ASAG systems, examining 80 papers published between 1996 and 2014. Their primary focus was on the progress of methods and approaches in this field. Their analysis indicated five eras of ASAG, including concept mapping [21], Information extraction [22], corpus-based [23], Machine learning [6,24], and evaluation methods (ASAP - SAS -Automatic Student Assessment Prize, RTE, SemEval Task, etc.). Fig. 1 shows the categorization of ASAG systems. Whereas the authors of [25] performed an analysis of 125 studies from 2016 to 2020 that explored the effects of automatic scoring and feedback in education.

Few works treated ASAG as a Classification task. In the study [6] the models for ASAG were designed using student and domain/question data and employing 31 features. Compared Deep Belief Networks (DBN) to other classifiers but faced limitations such as labor-intensive Knowledge Component (KC) extraction and binary outcome representation. Whereas the authors of [8] introduced “ans2vec,” employing skip-thought embedding for similarity measurement between student and reference answers.

The authors of [9] introduced a stacking model combining XGBOOST and a Neural Network with features from FASTTEXT sentence embedding and response word count for classification in short answer scoring. Authors of the study [5,26] approached ASAG as a multiclass classification challenge, treating each score as a distinct label. Eight similarities between the model and student responses were leveraged as features to construct classification models, including KNN, Naïve Bayes, SVM, Decision Tree, Random Forest, and XGBoost.

A few works that treated ASAG as a Regression task are as follows [27]. proposed GradeAid, an ASAG framework, utilizes the Term Frequency-Inverse Document Frequency (TF-IDF) and BERT Cross encoder semantic features to train regression models (SVR, RF, EN, RR, Ada), achieving an RMSE as low as 0.25 across datasets, including STITA. Whereas, in the study [28] they proposed an Auto

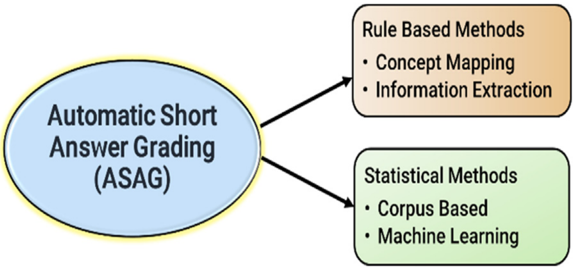

Fig. 1. Classification of automatic short answer grading (ASAG) methods [2].

SAS system that enhances existing features and introduces new ones, using a Random Forest regression model for short answer scoring. It incorporates Word2Vec, Doc2Vec, word frequency, difficulty levels, unique words, sentence and word length statistics, logical operator-based features, and temporal features. The authors of the study [29] introduced an automated method that generates text patterns with limited human effort and is generalizable across various datasets. They evaluated ASAP-SAS and Mohler/Texas datasets. On the ASAP dataset, they achieved a mean QWK of 0.78 compared to the existing 0.77. On the Mohler dataset, they correlated 0.61 and RMSE 0.77.

In the literature, some works have used deep learning methods for ASAG. For example, Recurrent Neural Networks [30], CNN [31], BERT [32–34]. Although deep learning models are widely applicable to ASAG tasks, their main limitation is the need for extensive training data and computational resources. We suggest a framework employing machine learning rather than deep learning, which performs effectively with limited data and requires fewer computational resources.

Second, the literature lacks methods for identifying inconsistency in student answer evaluation. The authors of the study [14] proposed an approach to identify outliers through the relationship between scores and symbolic markers or opinionated words. They contrasted scores and marker counts with peer answers lacking outliers but sharing similar scores. A limitation is the inability to detect inconsistencies without explicit indicators like ticks, crosses, or opinionated words.

Third, few works from the literature were found regarding the feedback on their performance in the exams to the students [23,24]. To our knowledge, no systems provide feedback to the evaluator regarding the inconsistent evaluation of student answers.

Except in Ref. [27] the validation experiments conducted in previous literature are constrained since authors commonly do not segregate datasets based on individual questions. Even when addressing the same subject, two questions might necessitate entirely distinct answers both lexically and semantically. Alternatively, some studies employed simplistic approaches, such as merely dividing datasets into training and testing sets and working only on a single dataset. There is a lack of consensus about the metrics to employ, the specific experiments to conduct, and the methods for carrying out these experiments.

3. Proposed methodology

In this section, we provide foundational information crucial for understanding subsequent sections. We detail the datasets used (Section 3.1), data preprocessing techniques (Section 3.2), methods for representing student answers/feature extraction (Section 3.3), experimented machine learning methods (Section 3.4), performance assessment metrics (Section 3.5), and the proposed approach for inconsistency detection and feedback (Section 3.6). Fig. 2 illustrates the *IntelliGrader* framework, aiding evaluators in three critical tasks: automatic grading of short descriptive answers, inconsistency identification, and providing feedback.

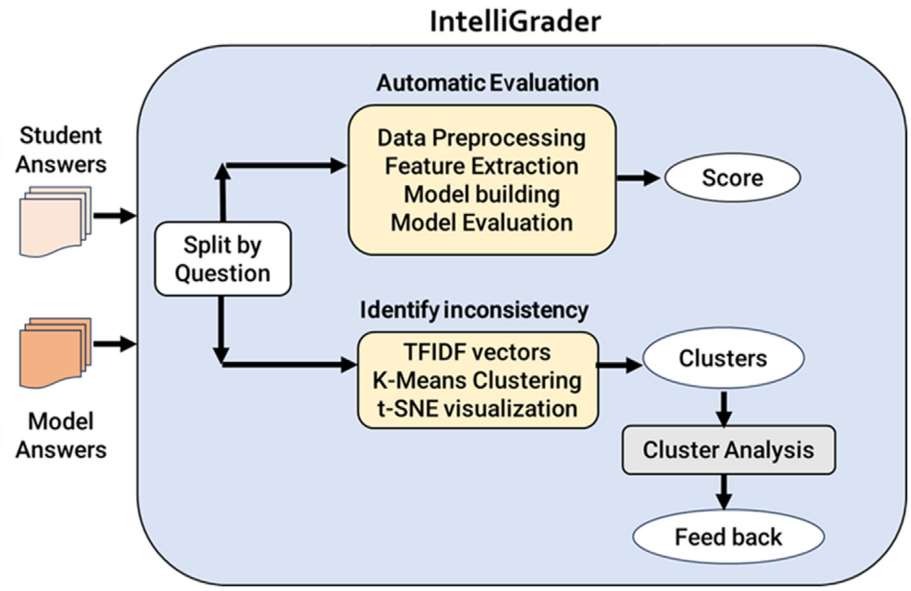

Fig. 2. IntelliGrader*: A framework for automatic grading of short answers, identifying evaluation inconsistency, and feedback.*

3.1. Dataset

The datasets mentioned in Table 1 were exploited in the proposed work to design the *IntelliGrader* system. Below is a detailed description of each dataset.

- **Automatic Student Assessment Prize-Short Answer Scoring (ASAP-SAS):** The Kaggle ASAP-SAS dataset, sponsored by the Hewlett Foundation, features ten prompts covering diverse subjects, each with around 2200 answers scored by human annotators.
- **STITA:** It is an Italian dataset, but GradeAid [27] made the English version of it publicly available. The link is given in Table 1. It consists of 333 samples of six questions from statistics.
- **IDEAS:** To overcome the limitations of the ASAG benchmark dataset, a new dataset, *IDEAS*, is introduced. This primary data is from the class VII social science curriculum from a school in Andhra Pradesh, India. Collected over six months from various exams, it comprises 800 answers across 20 questions, encompassing one-mark, two-mark, and four-mark questions.

3.2. Data pre-processing

Data preprocessing is performed before building the machine learning models to improve data quality. It includes removing punctuation/special characters, tokenization, stop words, stemming, and lemmatization.

3.3. Feature extraction

Eight similarities/features were extracted between Student Answers (SA) and Model Answers (MA). All are implemented using modules like Countvectorizer, Tfidfvectorizer, etc., from the Scikitlearn library using Python.

3.3.1. Statistical similarity

It offers insights into answer thoroughness and clarity by collecting statistical data from model and student responses. Metrics include sentence count, word count, and unique word usage, represented as vectors for similarity assessment. Euclidean distance measures statistical resemblance between student and model response vectors [1,24].

3.3.2. Word-word/BoW (bag of words) similarity

It aids in evaluating adherence to instructions, terminology usage, and definition precision. Using a count vectorizer, Bag of Words representations are created for student and model responses, with cosine similarity computed while considering stop words for comparison [1,24].

3.3.3. Keywords/unique words similarity

This aids instructors in assessing content relevance and identifying misconceptions via keyword usage. Unique word similarity computation excludes the stop words. The “stop word = ‘English’” parameter enhances the representation of the Bag of Words, enabling cosine similarity determination.

3.3.4. Lemmatized words similarity

Lemmatization identifies root forms in student and model responses, and then a count vectorizer creates lemmatized word collections, enabling cosine similarity computation. This enhances matching accuracy, efficiently identifying semantically similar terms and ensuring precise comparison, which is particularly beneficial for answers with grammatical variation.

3.3.5. TF-IDF similarity

The limitation of Bag of Words vectors is their incapacity to preserve details regarding an extensive vocabulary, word sequence, or sentence configuration. TfidfVectorizer generates Term Frequency-

Table 1. Exploited datasets in the IntelliGrader *framework.*

Dataset	Language	# Samples	# Questions	Each answer's average word length	Subject	Score/grade range	Limitations
ASAP[a]	English	17, 043	10	150−550	Varied: Biology, Arts, Science, English, etc.	N [0, 3]	More suits for the Automatic Essay Grading (AEG) systems. Does not suit model answer-based methods.
STITA[b]	Italian/but translated to English	333	6	50−100	Statistics	N [0, 1]	Suitable for Essay evaluation systems
IDEAS	English	800	20	10−50	Social Science	N [0,4]	–

[a] https://www.kaggle.com/c/asap-sas.

[b] https://github.com/edgresearch/datasetautomaticgrading-2022.

Inverse Document Frequency (TF-IDF) vectors for student and model responses to overcome this. Subsequently, the cosine similarity between these vectors is calculated.

3.3.6. Contextual similarity

TFIDF or Bag of Words vectors fail to capture the context of a sentence. To address this issue, Latent Semantic Analysis (LSA) is employed. Initially, the text of student and model responses is represented as TFIDF vectors, which are then subjected to dimensionality reduction using Singular Value Decomposition (SVD). Finally, the similarity is determined by evaluating the dot product of these reduced vectors. It effectively addresses paraphrases and synonyms, which is crucial for assessing answers with ambiguous language, particularly in scenarios with multiple-question interpretations [1,24].

3.3.7. Semantic similarity

It helps assess how well a student's answer grasps fundamental concepts from the reference answer, providing insights into their understanding. Higher semantic similarity between student and reference answers indicates better comprehension and aids in paraphrasing detection. Infersent [20] generates sentence embeddings for student and model responses, followed by the computation of cosine similarity between them.

3.3.8. Summary similarity

Extractive summarization generates summaries of Model and Student answers, followed by cosine similarity calculation, which aids in assessing subject comprehension. It assists in evaluating comprehension, identifying discrepancies and misinterpretations in students' understanding, and enables targeted feedback for improvement.

3.4. Machine learning method

Here, the ASAG task is considered a supervised learning regression problem. It utilizes regressors to assign scores to student answers by mapping the model and student responses. Eight extracted similarities between model and student answers are independent features, while evaluator-assigned scores are dependent features. Five state-of-the-art regressors are employed, including AdaBoost, Elastic Net, SVR, Random Forest, and Ridge Regression. These are chosen for their widespread use and availability in the scikit-learn Python library.

3.5. Evaluation metrics

The following evaluation metrics are used for model evaluation. Here, y_i is the actual evaluator score, $\hat{y_i}$ is the model predicted score and N is the number of samples.

i) Mean Absolute Error (MAE): It is the average absolute difference between actual and predicted values. The mathematical formula is given as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|y_i - \hat{y_i}\right| \tag{1}$$

ii) Mean Square Error (MSE): It is the average square of the differences between the actual and predicted scores.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{y_i}\right)^2 \tag{2}$$

iii) Root Mean Square Error (RMSE): It represents the square root of MSE.

$$RMSE = \sqrt{MSE} \tag{3}$$

iv) Normalized Root Mean Square Error (NRMSE): It scales the RMSE by the range of the data and provides a normalized measure facilitating comparison across the datasets with diverse scales.

$$NRMSE = \frac{RMSE}{Range\ of\ data} \tag{4}$$

3.6. Proposed approach to inconsistency check and feedback

Figure 3 depicts the proposed method for detecting intra-rater inconsistency in evaluations. The proposed work utilizes unsupervised learning through K-Means for clustering, TF-IDF for text representation, and t-distributed stochastic Neighbor Embedding (t-SNE) for visualization. It emphasizes cases where positive values are affected, like marking correct answers as incorrect, and can applied to any dataset. Initially, K-Means clustering groups answers for each question, and then TF-IDF quantifies word significance. The elbow method identifies optimal cluster numbers, and t-SNE visualizes high-dimensional clusters. Compared to Principal Component Analysis (PCA) and Singular Value Decomposition

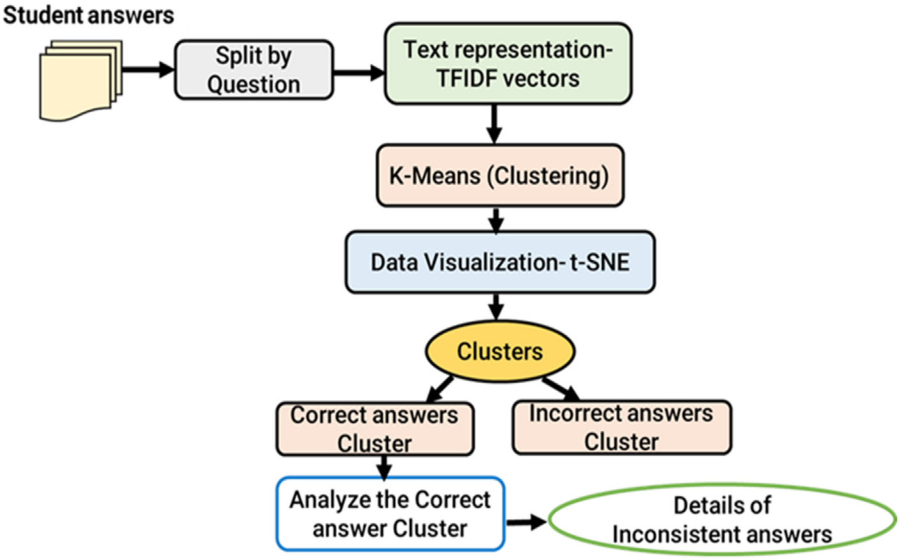
Student answers
Split by Question
Text representation-TFIDF vectors
K-Means (Clustering)
Data Visualization- t-SNE
Clusters
Correct answers Cluster
Incorrect answers Cluster
Analyze the Correct answer Cluster
Details of Inconsistent answers

Fig. 3. The proposed method for inconsistency check in evaluation and providing feedback.

(SVD), t-SNE effectively captures relationships, local structures, and non-linearities.

The t-SNE visualization displays two distinct clusters: one for accurately marked student answers and another for incorrect responses. We prioritize identifying inconsistencies within the correct cluster, focusing on answers erroneously labeled as incorrect, even though they are accurate, rather than concentrating on clusters of incorrect answers. This approach affords students the benefit of the doubt for precise potential responses. Post-analysis of the correct answer cluster, evaluators receive a detailed feedback report with Student ID, reference, and Student answer keywords encompass cosine, statistical, semantic, and summary similarities. This report aids in detecting assessment patterns and inconsistencies, facilitating corrective actions, and fostering impartial evaluations.

4. Results and discussion

This section discusses the vital experimental results attained after applying the *IntelliGrader* approach to various ASAG datasets. Section 4.1 discusses the experimental setup for implementing the proposed approach. Section 4.2 confers the results regarding the automatic scoring of short descriptive answers on numerous datasets. Section 4.3 deliberates on the comparative analysis of *IntelliGrader* and the existing literature work GradeAid results. Section 4.4 discusses the inconsistency checks results. Finally, Section 4.5 discusses the results regarding the feedback on inconsistent answers to the evaluator.

4.1. Experimental setup

Experiments, including preprocessing, feature extraction, and model building for one ASAP-SAS dataset question, were completed in under 15 min on an Intel Core i5-1035G1 CPU, 8 GB RAM, and a 64-bit OS. *IntelliGrader*'s GPU-free design democratizes access to machine learning, benefitting students, teachers, and schools by eliminating the need for high-powered systems. Machine learning aids result in interpretability and facilitate feedback provision on inconsistent answers.

4.2. Results concerning automatic short answer grading on various datasets

Tables 2–4 show the results of the proposed methodology for ASAG on the STITA, ASAP-SAS, and novel dataset IDEAS, respectively. These tables depict the MAE, NRMSE, and RMSE results attained by all the regressors AdaBoost, Elastic Net, SVR, Random Forest, and Ridge Regression for every question in each dataset (split by question). The lower values in the tables indicate the best fit of the methods. The least values are represented in bold.

The ASAP-SAS dataset is sampled using a 10-fold cross-validation as the large dataset has 1672 answers for each of the ten questions. The best achieved MAE is 0.27, NRMSE is 0.18, and RMSE is 0.53 for Set 6. For all ten sets of questions, the Random forest regressor gave promising results compared to all other regressors concerning all regression metrics. This is mainly due to the vital features of

Table 2. Results attained using LOOCV on the STITA dataset (5 Questions).

Regressor	Q1			Q2			Q3			Q4			Q5		
	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE
AdaBoost	0.11	0.15	0.15	0.13	0.17	0.17	0.07	0.11	0.12	0.12	0.17	0.17	**0.19**	**0.25**	**0.25**
Elastic Net	**0.08**	**0.11**	**0.11**	0.14	0.19	0.19	0.07	0.10	0.11	0.16	0.19	0.19	0.23	0.28	0.28
SVR	0.09	0.12	0.12	0.14	0.19	0.19	0.07	0.10	0.11	0.11	0.14	0.14	0.22	0.27	0.27
Random Forest	0.09	0.14	0.14	0.13	0.18	0.18	0.06	0.09	0.10	**0.10**	**0.15**	**0.15**	0.20	0.27	0.27
Ridge Regression	0.09	0.12	0.12	**0.11**	**0.14**	**0.14**	**0.06**	**0.08**	**0.09**	0.14	0.17	0.17	0.21	0.26	0.26

Bold values helps to identify the best results clearly. This shows the significant comparison between the existing work and the proposed work.

Table 3. Results attained using 10-fold cross-validation on the ASAP dataset of 10 questions (Sets).

Regressor	SET 1			SET 2			SET 3			SET 4			SET 5		
	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE
AdaBoost	0.80	0.95	**0.32**	0.78	0.93	**0.31**	**0.45**	**0.58**	**0.29**	0.45	0.54	**0.27**	0.42	0.49	0.16
Elastic Net	0.85	1.00	0.33	0.83	0.98	0.33	0.48	0.60	0.30	0.46	0.57	0.28	0.40	0.58	0.19
SVR	0.82	1.02	0.34	0.79	0.99	0.33	0.45	0.62	0.31	0.42	0.57	0.29	0.33	0.57	0.19
Random Forest	**0.78**	**0.95**	**0.32**	**0.76**	**0.93**	**0.31**	**0.46**	**0.59**	**0.30**	0.44	0.55	**0.27**	**0.26**	**0.43**	**0.14**
Ridge Regression	0.79	0.95	0.32	0.77	0.93	0.31	0.46	0.58	0.29	0.44	0.55	0.27	0.35	0.50	0.17
Regressor	SET 6			SET 7			SET 8			SET 9			SET 10		
	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE
AdaBoost	0.42	0.60	0.20	0.70	0.80	0.40	0.65	0.75	0.37	**0.51**	**0.61**	**0.30**	0.49	0.58	0.29
Elastic Net	0.38	0.63	0.21	0.72	0.80	0.40	0.71	0.81	0.41	0.57	0.73	0.36	0.49	0.65	0.33
SVR	0.32	0.67	0.22	0.68	0.85	0.43	0.66	0.84	0.42	0.52	0.69	0.34	0.44	0.63	0.32
Random Forest	**0.27**	**0.53**	**0.18**	**0.67**	**0.79**	**0.40**	**0.63**	**0.76**	**0.38**	0.51	0.62	0.31	**0.46**	**0.59**	**0.29**
Ridge Regression	0.39	0.60	0.20	0.70	0.80	0.40	0.64	0.75	0.37	0.55	0.67	0.33	0.48	0.61	0.30

Bold values helps to identify the best results clearly. This shows the significant comparison between the existing work and the proposed work.

Table 4. Results attained using LOOCV on the IDEAS dataset (20 Questions).

Regressor	Q1			Q2			Q3			Q4			Q5		
	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE
AdaBoost	0.42	0.51	0.51	0.18	0.41	0.41	0.45	0.59	0.59	0.07	0.27	0.27	0.21	0.45	0.45
Elastic Net	0.51	0.51	0.51	0.31	0.36	0.36	0.51	0.51	0.51	0.16	0.24	0.24	0.31	0.38	0.38
SVR	0.44	0.53	0.53	0.24	0.32	0.32	**0.38**	**0.48**	**0.48**	**0.11**	**0.22**	**0.22**	0.26	0.39	0.39
Random Forest	**0.38**	**0.49**	**0.49**	**0.16**	**0.29**	**0.29**	0.38	0.51	0.51	0.10	0.26	0.26	**0.20**	**0.33**	**0.33**
Ridge Regression	0.43	0.47	0.47	0.26	0.32	0.32	0.50	0.52	0.52	0.15	0.23	0.23	0.26	0.35	0.35
Regressor	Q6			Q7			Q8			Q9			Q10		
	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE
AdaBoost	0.13	0.30	0.30	0.46	0.71	0.36	**0.31**	**0.51**	**0.25**	**0.26**	**0.47**	**0.24**	0.32	0.54	0.27
Elastic Net	0.48	0.50	0.50	0.69	0.76	0.38	0.69	0.78	0.39	0.42	0.55	0.27	0.36	0.48	0.24
SVR	0.37	0.48	0.48	0.56	0.73	0.36	0.76	0.86	0.43	0.35	0.53	0.27	0.29	0.44	0.22
Random Forest	**0.19**	**0.19**	**0.19**	**0.44**	**0.64**	**0.32**	0.35	0.52	0.26	0.32	0.47	**0.24**	**0.34**	**0.47**	**0.24**
Ridge Regression	0.27	0.34	0.34	0.55	0.63	0.32	0.45	0.53	0.26	0.37	0.51	0.26	0.36	0.49	0.24
Regressor	Q11			Q12			Q13			Q14			Q15		
	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE
AdaBoost	0.31	0.55	0.27	0.35	0.57	0.28	0.54	0.74	0.18	0.82	1.09	0.27	0.41	0.71	0.18
Elastic Net	0.41	0.55	0.27	0.44	0.54	0.27	0.61	0.73	0.18	1.34	1.60	0.40	0.90	1.05	0.26
SVR	**0.30**	**0.54**	**0.27**	0.42	0.53	0.27	0.56	0.69	0.17	1.10	1.41	0.35	0.66	0.81	0.20
Random Forest	0.36	0.57	0.29	**0.35**	**0.49**	**0.25**	**0.47**	**0.63**	**0.16**	**0.77**	**1.00**	**0.25**	**0.46**	**0.64**	**0.16**
Ridge Regression	0.38	0.54	0.27	0.44	0.54	0.27	0.51	0.60	0.15	0.87	1.10	0.27	0.73	0.85	0.21
Regressor	Q16			Q17			Q18			Q19			Q20		
	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE	MAE	RMSE	NRMSE
AdaBoost	0.25	0.50	0.13	0.47	0.72	0.18	0.41	0.65	0.16	0.87	1.22	0.30	0.44	0.52	0.13
Elastic Net	0.42	0.56	0.14	0.64	0.77	0.19	0.37	0.50	0.12	1.24	1.40	0.35	0.40	0.48	0.12
SVR	0.45	0.57	0.14	0.68	0.82	0.20	0.38	0.49	0.12	1.44	1.66	0.41	0.39	0.47	0.12
Random Forest	**0.30**	**0.48**	**0.12**	**0.44**	**0.69**	**0.17**	0.38	0.53	**0.13**	**0.71**	**1.03**	**0.26**	**0.35**	**0.47**	**0.12**
Ridge Regression	0.37	0.51	0.13	0.61	0.75	0.19	**0.35**	**0.48**	**0.12**	0.87	1.17	0.29	0.39	0.47	0.12

Bold values helps to identify the best results clearly. This shows the significant comparison between the existing work and the proposed work.

random forests, like robustness to the outliers and the capability to capture complex relationships. Concerning the STITA dataset, no particular regressor is performing well on sets. The ridge regression for Sets 2 and 3 showed promising results with the least NRMSE of 0.09. Concerning the IDEAS dataset, again, the random forest regressor achieved promising results with the lowest NRMSE of 0.12.

4.3. Comparative analysis of IntelliGrader and GradeAid results

After applying the proposed IntelliGrader approach to various benchmark and novel datasets, we compared the results with those of existing methods, namely GradeAid [27] from the literature. The MAE and RMSE values are not definitive indicators of the best fit, as their maximum values vary depending on the specific dataset. However, NRMSE is independent of scale and aids in comparing various datasets. So, we have compared the NRMSE values of the proposed approach *IntelliGrader* with GradeAid on multiple datasets. Tables 5 and 6 depict the comparative analysis results of NRMSE values across ASAP-SAS and STITA datasets. Experimental results show *IntelliGrader* outperforms the GradeAid results concerning all datasets with the lowest NRMSE values question-wise. This shows the importance of the eight features extracted from the model and the student answers, whereas, in *Grade Aid's* work, they extracted only two features, TF-IDF and semantic features, using BERT. The results prove a drastic improvement in NRMSE values by including these additional features for automatic scoring. The least values are represented in bold.

4.4. Results concerning the inconsistency check and feedback on various datasets

Here, we examine the outcomes related to detecting inconsistencies in evaluation. Inconsistencies are identified question-wise in each dataset. We treat an answer as inconsistent if it is marked as incorrect in a correct cluster.

The proposed inconsistency check and feedback methodology is initially applied to the IDEAS dataset. Figs. 4–7 illustrate the sample clusters for two IDEAS dataset questions, showing incorrect (0) and correct (1) clusters for actual and predicted marks. Question 1 has five inconsistent answers in actual marks and 8 in predicted marks, while question 2 exhibits one inconsistency. Fig. 8 compares the inconsistent answers for actual and predicted marks for all 20 IDEAS dataset questions.

Table 5. Comparison of NRMSE values between GradeAid and IntelliGrader on the ASAP dataset.

Regressor	SET 1		SET 2		SET 3		SET 4		SET 5	
	Grade Aid	*IntelliGrader*	GradeAid	*IntelliGrader*	GradeAid	*IntelliGrader*	GradeAid	*IntelliGrader*	GradeAid	*IntelliGrader*
AdaBoost	0.50	**0.32**	0.57	**0.31**	0.48	**0.29**	0.69	**0.27**	1.39	**0.16**
Elastic Net	0.52	**0.33**	0.45	**0.33**	0.54	**0.30**	0.78	**0.28**	1.79	**0.19**
SVR	0.42	**0.34**	0.4	**0.33**	0.44	**0.31**	0.58	**0.29**	1.35	**0.19**
Random Forest	0.44	**0.32**	0.42	**0.31**	0.45	**0.30**	0.59	**0.27**	1.30	**0.14**
Ridge Regression	0.52	**0.32**	0.45	**0.31**	0.54	**0.29**	0.77	**0.27**	1.77	**0.17**
Regressor	SET 6		SET 7		SET 8		SET 9		SET 10	
	Grade Aid	*IntelliGrader*	GradeAid	*IntelliGrader*	GradeAid	*IntelliGrader*	GradeAid	*IntelliGrader*	GradeAid	*IntelliGrader*
AdaBoost	2.11	**0.20**	0.97	**0.40**	0.61	**0.37**	0.47	**0.30**	0.45	**0.29**
Elastic Net	2.23	**0.21**	1.01	**0.40**	0.65	**0.41**	0.49	**0.36**	0.46	**0.33**
SVR	1.59	**0.22**	0.88	**0.43**	0.57	**0.42**	0.41	**0.34**	0.38	**0.32**
Random Forest	1.52	**0.18**	0.88	**0.40**	0.58	**0.38**	0.41	**0.31**	0.40	**0.30**
Ridge Regression	2.26	**0.20**	1.02	**0.40**	0.64	**0.37**	0.49	**0.33**	0.46	**0.30**

Bold values helps to identify the best results clearly. This shows the significant comparison between the existing work and the proposed work.

Table 6. Comparison of NRMSE values between GradeAid and IntelliGrader on the STITA dataset.

Regressor	Q1		Q2		Q3		Q4		Q5	
	Grade Aid	*IntelliGrader*	Grade Aid	*IntelliGrader*	Grade Aid	*IntelliGrader*	Grade Aid	*IntelliGrader*	Grade Aid	*IntelliGrader*
AdaBoost	0.19	**0.15**	0.17	**0.17**	0.17	**0.12**	0.23	**0.17**	0.54	**0.25**
Elastic Net	0.21	**0.11**	0.40	**0.19**	0.30	**0.11**	0.32	**0.19**	0.53	**0.28**
SVR	0.21	**0.12**	0.25	**0.19**	0.25	**0.11**	0.30	**0.14**	0.49	**0.27**
Random Forest	0.21	**0.14**	0.19	**0.18**	0.22	**0.10**	0.27	**0.15**	0.36	**0.27**
Ridge Regression	0.21	**0.12**	0.33	**0.14**	0.22	**0.09**	0.28	**0.17**	0.45	**0.26**

Bold values helps to identify the best results clearly. This shows the significant comparison between the existing work and the proposed work.

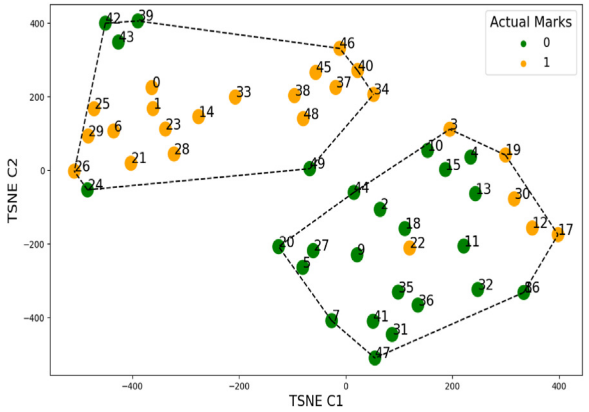

Fig. 4. Actual marks for Question 1 from the IDEAS dataset depicted using t-SNE.

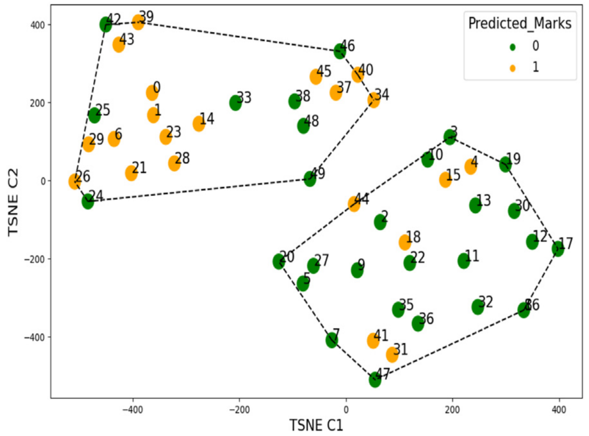

Fig. 5. Random Forest regression model predicted marks for Question 1 from the IDEAS dataset depicted using t-SNE.

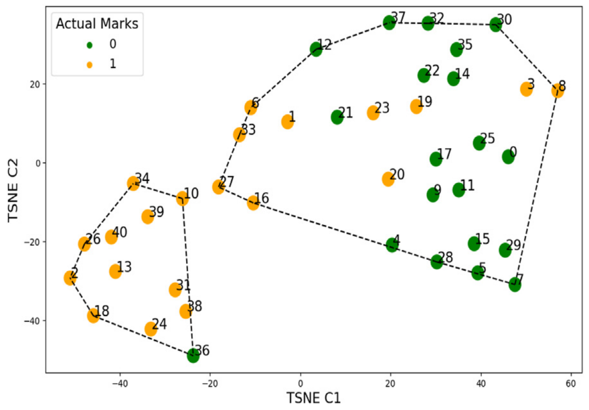

Fig. 6. Actual marks for Question 2 from the IDEAS dataset depicted using t-SNE.

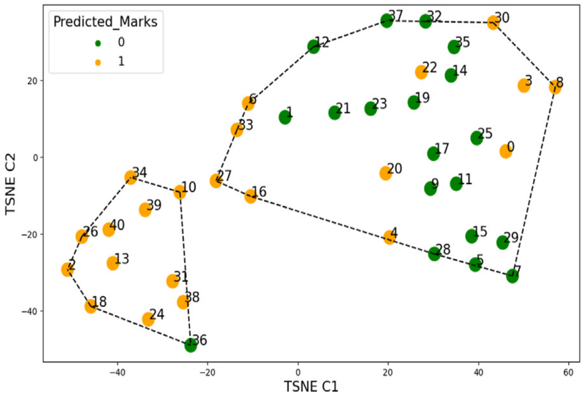

Fig. 7. Random Forest regression model predicted marks for Question 2 from the IDEAS dataset depicted using t-SNE.

Results depict that out of 800 answers, 108 are inconsistent with actual marks and 107 with the regression model predicted marks. Likewise, Fig. 9 compares the inconsistent answers for the six questions in the STITA dataset. Experimental findings indicate that out of 333 answers, 19 were inconsistently evaluated concerning the evaluator marks and the regression model's predicted marks. Results indicate that the model's predicted scores are equivalent to or sometimes less inconsistent than the scores provided by human evaluators. This underscores the model's performance is on par with and sometimes beats the human evaluator's.

4.5. Results concerning the feedback on inconsistent evaluation to the evaluator

Once the inconsistent answers are identified, detailed feedback concerning inconsistently evaluated answers is given to the evaluator. This feedback includes student ID, model/reference answer keywords, student answer keywords, and details regarding how much the student's answer matches the model answer in terms of similarities like cosine similarity, statistical similarity, Semantic Similarity, Summary Similarity, actual evaluator marks, and marks predicted by the system. In this representation, matched keywords in student and model answers are highlighted in green, while mismatched keywords are indicated in red. Table 7 depicts the sample feedback details of inconsistent answers of actual marks to IDEAS dataset question 1.

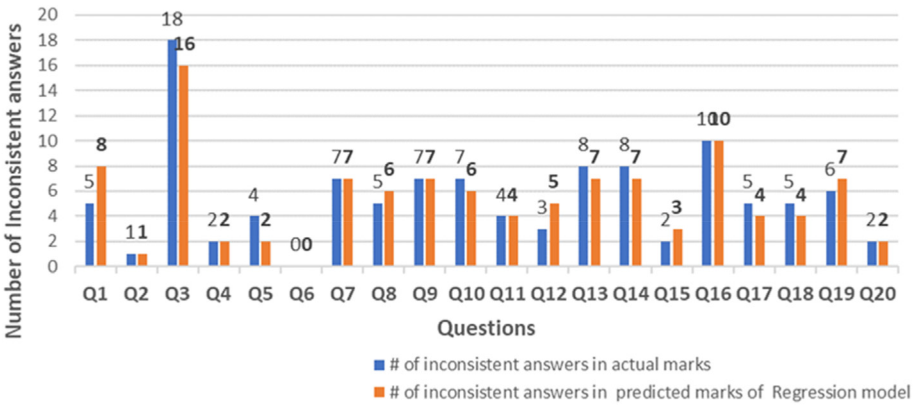

Fig. 8. Comparison of the number of inconsistently evaluated answers between actual and regressor model predicted scores in the IDEAS dataset (20 questions).

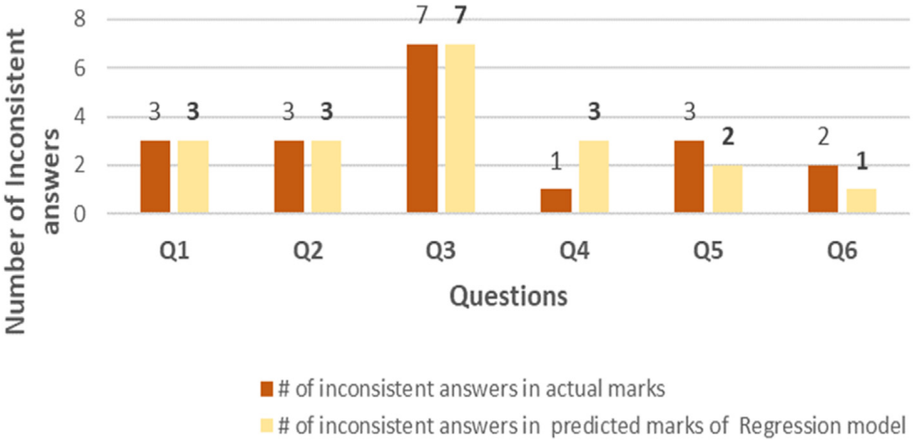

Fig. 9. Comparison of the number of inconsistently evaluated answers between actual and regressor model predicted scores for the STITA dataset (6 Questions).

Table 7. Feedback details for IDEAS dataset Question.

STD ID	Model Answer Keywords	Student Answer Keywords	Cosine	Statistical	Semantic	Summary	Actual Marks	Predicted Marks
36	matches, one, one, many, 1.save, climate, keep, rare ,we, million, care, tree, 3., match, forests, save, 2., now, they, future.4., save, destroy, make	tree, make, match, one	0.361	55.145	0.92	0.44	0	0

5. Conclusion

The presented work addresses three critical issues of the Automatic Short Answer Grading (ASAG) realm i.e., automatic grading of short subjective answers, finding the inconsistency in the evaluation, and providing complete feedback concerning inconsistency to the evaluator. A new framework for ASAG, *IntelliGrader*, is proposed involving eight varieties of features/similarities that gauge all aspects of likenesses between model and student answers. To benchmark the performance of the anticipated *IntelliGrader*, it is validated on various publicly available datasets like ASAP-SAS, STITA, and novel dataset IDEAS. Comparative analysis with existing literature GradeAid showed that the proposed framework is beneficial in the case of all datasets. An approach to identify intra-rater inconsistency in the evaluation is proposed. The comparative analysis of inconsistency in actual and predicted marks proved that the model-predicted scores are less inconsistent than the actual marks—finally, detailed feedback regarding the inconsistent answers is provided to the evaluator. The proposed system is not a replacement for human evaluation but a screening tool that aids evaluators in performing evaluation tasks.

Despite the hopeful results of *IntelliGrader*, the proposed framework has a few limits. Here, we used regressors for scoring that assign the real scores, so further discriminator is needed to convert the real number to an integer value. This can be resolved using an ordinal regressor. Further, syntactic features can be included using subtree kernels and existing features in the future. In the future, explainable AI can be incorporated for results interpretation Cheating/security aspects of the automatic scoring systems should also be addressed. We can extend the work by using deep learning models for scoring and analyzing the advantages and disadvantages of the current *IntelliGrader*. The ultimate goal is to create an *IntelliGrader* API that reads voice input and scores accordingly.

Ethics information

Not applicable.

Funding

This research did not receive any specific funding from public, commercial, or not-for-profit sectors.

Acknowledgement

The authors express their gratitude to REVA University for providing the facilities and support essential for conducting this research.

References

[1] S.K. Saha, R. Gupta, Adopting computer-assisted assessment in evaluation of handwritten answer books: an experimental study, Educ Inf Technol 25 (2020) 4845–4860, https://doi.org/10.1007/s10639-020-10192-6.

[2] S. Burrows, I. Gurevych, B. Stein, The eras and trends of automatic short answer grading, Int J Artif Intell Educ 25 (2015) 60–117, https://doi.org/10.1007/s40593-014-0026-8.

[3] P.S. Lakshmi, Kavitha, an improved hybrid stacked classifier for multi label text categorization, Int J Recent Technol Eng 8 (2019) 5911–5915, https://doi.org/10.35940/ijrte.C4739.098319.

[4] K.F.S. Hall, Multiple-choice exams: an obstacle for higher-level thinking in introductory science classes, CBE-Life Sci Educ 11 (2012) 294–306, https://doi.org/10.1187/cbe.11-11-0100.

[5] P.S. Lakshmi, J.B. Simha, A hybrid qualitative and quantitative approach for automatic short answer grading using classification algorithms, in: 2022 4th International Conference on Circuits, Control, Communication, and Computing (I4C), IEEE. 2022, pp. 12–17, https://doi.org/10.1109/I4C57141.2022.10057906.

[6] Y. Zhang, C. Lin, M. Chi, Going deeper: automatic short-answer grading by combining student and question models, User Model User-Adapted Interact 30 (2020) 51–80, https://doi.org/10.1007/s11257-019-09251-6.

[7] W.J. Hou, J.H. Tsao, S.Y. Li, L. Chen, Automatic assessment of students' free-text answers with support vector machines, Lect Notes Comput Sci, Springer 6096 (2010) 235–243, https://doi.org/10.1007/978-3-642-13022-9_24.

[8] W.H. Gomaa, A.A. Fahmy, Ans2vec: a scoring system for short answers, in: Advances in Intelligent Systems and Computing, Springer Verlag. 2020, pp. 586–595, https://doi.org/10.1007/978-3-030-14118-9_59.

[9] R.A. Rajagede, R.P. Hastuti, Stacking neural network models for automatic short answer scoring, IOP Conf Ser Mater Sci Eng 1077 (2021) 012013, https://doi.org/10.1088/1757-899x/1077/1/012013.

[10] J. Mueller, A. Thyagarajan, Siamese recurrent architectures for learning sentence similarity, Proc AAAI Conf Artif Intell 30 (2016) 1 2786–2792, https://doi.org/10.1609/aaai.v30i1.10350.

[11] U. Orhan, C.N. Tulu, A novel embedding approach to learn word vectors by weighting semantic relations: SemSpace, Expert Syst Appl 180 (2021) 115146, https://doi.org/10.1016/j.eswa.2021.115146.

[12] A. Prabhudesai, T.N.B. Duong, Automatic short answer grading using siamese bidirectional LSTM based regression, in: IEEE International Conference on Engineering, Technology, and Education (TALE), Yogyakarta, Indonesia, 2019, pp. 1–6, https://doi.org/10.1109/TALE48000.2019.9226026.

[13] M. Beseiso, S. Alzahrani, An empirical analysis of BERT embedding for automated essay scoring, Int J Adv Comput Sci Appl 11 (2020) 204–210, https://doi.org/10.14569/IJACSA.2020.0111027.

[14] A. Shukla, B.D. Chaudhary, A strategy for detection of inconsistency in evaluation of essay type answers, Educ Inf Technol 19 (2014) 899–912, https://doi.org/10.1007/s10639-013-9264-x.

[15] J.R.R. Juan, A.J. Gallego, J.C. Zaragoza, Automatic detection of inconsistencies between numerical scores and textual feedback in peer-assessment processes with machine learning, Comput Educ 140 (2019) 103609, https://doi.org/10.1016/j.compedu.2019.103609.

[16] J.P. Bernius, S. Krusche, B. Bruegge, Machine learning based feedback on textual student answers in large courses, Comput Educ: Artif Intell 3 (2022) 100081, https://doi.org/10.1016/j.caeai.2022.100081.

[17] A.V.Y. Lee, A.C. Luco, S.C. Tan, A human-centric automated essay scoring and feedback system for the development of ethical reasoning, Technol Soc 26 (2023) 147–159, https://doi.org/10.30191/ETS.202301_26(1).0011.

[18] Q. Hao, D.H. Smith, L. Ding, A. Ko, C. Ottaway, J. Wilson, K.H. Arakawa, A. Turcan, T. Poehlman, T. Greer, Towards understanding the effective design of automated formative feedback for programming assignments, Comput Sci Educ 32 (2022) 105–127, https://doi.org/10.1080/08993408.2020.1860408.

[19] N. Suzen, A.N. Gorban, J. Levesley, E.M. Mirkes, Automatic short answer grading and feedback using text mining methods, in: Procedia Comput Sci, Elsevier B.V. 2020, pp. 726–743, https://doi.org/10.1016/j.procs.2020.02.171.

[20] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, A. Bordes, Supervised learning of universal sentence representations from natural language inference data, in: Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Copenhagen, Denmark. 2017, pp. 670–680, https://doi.org/10.18653/v1/D17-1070.

[21] C. Leacock, M. Chodorow, C-Rater: automated scoring of short-answer questions, Comput Humanit 37 (2003) 389–405, https://doi.org/10.1023/A:1025779619903.

[22] R. Siddiqi, C. Harrison, A systematic approach to the automated marking of short-answer questions, in: IEEE INMIC 2008: 12th IEEE International Multitopic Conference, 2008, pp. 329–332, https://doi.org/10.1109/INMIC.2008.4777758.

[23] E. Alfonseca, D. Perez, Automatic assessment of open ended questions with a bleu-inspired algorithm and shallow NLP, in: Advances in Natural Language Processing, 4th International Conference, EsTAL 2004, Alicante, Spain, 2004, pp. 25—35, https://doi.org/10.1007/978-3-540-30228-5_3.

[24] S.K. Saha, D.R. CH, Development of a practical system for computerized evaluation of descriptive answers of middle school level students, Interact Learn Environ 30 (2022) 215—228, https://doi.org/10.1080/10494820.2019.1651743.

[25] M.G. Hahn, S.M.B. Navarro, L.D.L.F. Valentin, D. Burgos, A systematic review of the effects of automatic scoring and automatic feedback in educational settings, IEEE Access 9 (2021) 108190—108198, https://doi.org/10.1109/ACCESS.2021.3100890.

[26] P.S. Lakshmi, J.B. Simha, R. Ranjan, Empowering educators: automated short answer grading with inconsistency check and feedback integration using machine learning, SN Comput Sci 5 (2024) 653, https://doi.org/10.1007/s42979-024-02954-7.

[27] E.D. Gobbo, A. Guarino, B. Cafarelli, L. Grilli, GradeAid: a framework for automatic short answers grading in educational contexts—design, implementation and evaluation, Knowl Inf Syst 65 (2023) 4295—4334, https://doi.org/10.1007/s10115-023-01892-9.

[28] Y. Kumar, S. Aggarwal, D. Mahata, R.R. Shah, P. Kumaraguru, R. Zimmermann, Get IT scored using AutoSAS -an automated system for scoring short answers, Proc AAAI Conf Artif Intell 33 (2019) 9662—9669, https://doi.org/10.1609/aaai.v33i01.33019662.

[29] L. Ramachandran, J. Cheng, P. Foltz, Identifying patterns for short answer scoring using graph-based lexico-semantic text matching, in: Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications, Association for Computational Linguistics, Colorado. 2015, pp. 97—106, https://doi.org/10.3115/v1/W15-0612.

[30] C. Cai, Automatic essay scoring with recurrent neural network, in: Proceedings of the 3rd International Conference on High-Performance Compilation, Computing, and Communications, ACM, New York, USA, 2019, pp. 1—7, https://doi.org/10.1145/3318265.3318296.

[31] Z. Chen, Y. Zhou, Research on automatic essay scoring of composition based on CNN and OR, in: 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD), IEEE. 2019, pp. 13—18, https://doi.org/10.1109/ICAIBD.2019.8837007.

[32] C. Sung, T. Dhamecha, S. Saha, T. Ma, V. Reddy, R. Arora, Pre-training BERT on domain resources for short answer grading, in: Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Stroudsburg, PA, USA. 2019, pp. 6070—6074, https://doi.org/10.18653/v1/D19-1628.

[33] H.A. Ghavidel, A. Zouaq, M.C. Desmarais, Using BERT and XLNET for the automatic short answer grading task, in: CSEDU 2020 - Proceedings of the 12th International Conference on Computer Supported Education, SciTePress. 2020, pp. 58—67, https://doi.org/10.5220/0009422400580067.

[34] F. Jamil, I.A. Hameed, Toward intelligent open-ended questions evaluation based on predictive optimization, Expert Syst Appl 231 (2023) 120640, https://doi.org/10.1016/J.ESWA.2023.120640.