An Automatic Grading System for Arabic Language Short-Answer Questions Using Deep Learning

Afnan Alqurashi
Computer Science and Artificial Intelligence Department, College of Computer Science and Engineering, University of Jeddah, Saudi Arabia
aalqurashi0168.stu@uj.edu.sa (corresponding author)

Basma Alharbi
Computer Science and Artificial Intelligence Department, College of Computer Science and Engineering, University of Jeddah, Saudi Arabia
bmalharbi@uj.edu.sa

Sahar Sabbeh
Information Science and Technology Department, College of Computer Science and Engineering, University of Jeddah, Saudi Arabia | Department of Information Systems, Faculty of Computer Science and Artificial Intelligence, Benha University, Egypt
sfsabbeh@uj.edu.sa

Received: 11 March 2025 | Revised: 30 April 2025 and 23 June 2025 | Accepted: 27 June 2025

 DOI: https://doi.org/10.48084/etasr.10917

ABSTRACT

Assessing students' acquired knowledge is a core objective of the educational process. Automating this task can improve the quality of the evaluation and reduce the time, effort, and cost associated with manual grading. Automated Short Answer Grading (ASAG) systems aim to support this goal by providing accurate scoring and clear feedback to students. The ability to explain assigned scores is crucial for the real-world deployment of automated systems, ensuring transparency and trust. This work introduces an Arabic ASAG system designed to combine automated scoring with explainability. Several deep learning models and embedding techniques are evaluated, including BERT, LSTM, and attention mechanisms. Experiments are conducted on three datasets: AR-ASAG, a dedicated Arabic ASAG dataset; a manually translated Arabic version of the PT-ASAG dataset (originally in Portuguese); and a merged dataset combining both. The results highlight the effectiveness of the BERT model, which achieved strong performance with Pearson correlation coefficients from 0.811 to 0.923 and a minimum RMSE of 0.182. Prediction results are also interpreted to improve both the explainability and reliability of the system.

Keywords-ASAG; short answer questions; Arabic language; short answer grading; deep learning; RNN; LSTM; BERT; explainable AI

I. INTRODUCTION

Since the onset of the COVID-19 pandemic in 2020, online education has become a necessity, leading educational institutions to rapidly adopt remote learning strategies. This shift not only transformed traditional pedagogical methods but also presented challenges in assessing student performance, particularly in large-scale educational environments. Multiple-Choice Questions (MCQs), favored for their ease of use and compatibility with automated grading tools [1], fail to capture higher-order thinking and deeper understanding [2] because they are limited to assessing recognition and recall abilities, offering little insight into critical thinking or problem-solving skills. In contrast, short-answer and essay questions provide a more comprehensive evaluation of student knowledge and skills, allowing for an assessment of deeper cognitive abilities such as analysis, reasoning, and synthesis [3]. However, manually grading such open-ended responses is time-consuming and costly, particularly in large educational settings such as MOOCs (Massive Open Online Courses).

To address these challenges, Automated Essay Scoring (AES) systems were developed. Early systems, such as Project Essay Grader (PEG), used basic linguistic features such as word length and sentence complexity to score essays [4]. The Intelligent Essay Assessor (IEA) used latent semantic analysis

to evaluate the meaning of essays, further advancing automated grading capabilities [5]. However, early systems struggled to capture the deeper semantic and content-based nuances required for accurate grading.

The integration of machine learning techniques in the early 2000s significantly improved AES systems. Feature-based models that extracted linguistic features such as grammar, coherence, and relevance allowed more accurate essay grading, utilizing large training corpora for better generalization [6]. However, these models required extensive feature engineering and often struggled with diverse writing styles. Since 2015, deep learning has transformed the field of Automated Short Answer Grading (ASAG). Techniques such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Long Short-Term Memory (LSTM) networks have been applied to automate the grading process, leading to notable advances in both accuracy and scalability [7]. Advanced embeddings have proven instrumental in the transformation of textual data into dense numerical vectors that represent both syntactic and semantic properties of language [8]. Word and sentence embedding techniques are employed to represent the relationships between words and phrases effectively.

- Word embedding is a Natural Language Processing (NLP) technique that transforms words into numerical vectors in a continuous space, capturing their semantic relationships based on contextual usage. This study utilizes two pre-trained word embedding models: GloVe and FastText. GloVe [9] generates word vectors by analyzing global word co-occurrence statistics and factorizing a co-occurrence matrix to reflect semantic patterns. On the other hand, FastText [10] enhances traditional word embedding by incorporating subword information through character n-grams, making it effective for handling rare and out-of-vocabulary words, especially in morphologically rich languages. Both models used in this work produce embeddings with 300 dimensions.
- Sentence embedding transforms entire sentences into fixed-size vectors that capture their semantic and contextual meaning. This study uses two models: the Universal Sentence Encoder [11], which produces 512-dimensional vectors and is trained to understand sentence-level meaning for tasks such as similarity and clustering, and Sentence Transformers [12], a framework that generates dense representations of sentences, enabling advanced applications such as semantic similarity, search, and paraphrase detection.

Despite significant advances in deep learning for automated grading, challenges related to transparency and interpretability remain unresolved. Although deep learning models excel in accuracy, their black-box nature provides little insight into the reasoning behind specific decisions. This opacity raises critical concerns in educational settings, where fairness, accountability, and trust are essential [13]. To address these issues, the emerging field of Explainable AI (XAI) focuses on developing methods that make AI systems more interpretable and transparent, thus increasing educators' confidence in the use of automated grading systems [14].

Automated grading systems have the potential to revolutionize education, particularly with ongoing advances in NLP and AI. However, their future lies not only in improving accuracy but also in addressing fairness, transparency, and ethical considerations. The integration of XAI and the development of interpretable models will play a pivotal role in fostering widespread acceptance and trust in these systems within educational environments. This study introduces an Automated Arabic Short Answer Grading (ASAG) system that prioritizes both interpretability and effectiveness. The methodology involves leveraging advanced deep learning models, including RNN, LSTM, and BERT, to evaluate student responses. This study experimented with multiple embedding techniques across three datasets: the AR-ASAG dataset, a manually translated Arabic version of the PT-ASAG dataset, and a combined dataset. The contributions of this study include the introduction of an explainable grading framework that interprets prediction outcomes, a comparative analysis of deep learning models for Arabic ASAG, and a demonstration of the superior performance of the BERT model, achieving Pearson Correlation Coefficients from 0.811 to 0.923 and a minimal RMSE of 0.182.

II. RELATED WORK

Various methods have been developed to automate short answer grading, falling into either text similarity techniques or AI-driven approaches. The first category treats the problem as a text-to-text similarity issue between the student's response and the rubric, exploring syntactic and/or semantic features [15]. The second category leverages AI techniques, including machine learning and deep learning models. Recent research has introduced robust models in the field of ASAG, particularly deep learning approaches such as RNN, LSTM [7], and BERT [8]. This study reviewed related studies on ASAG and the importance of explainability within ASAG systems. Section A describes state-of-the-art ASAG techniques, while Section B examines existing explainability efforts.

A. ASAG

Recent studies have explored the use of deep learning models in ASAG, highlighting diverse approaches that leverage embedding techniques and neural network architectures to improve grading accuracy and interpretability. In [16], the focus was on paragraph embedding methods combined with word embeddings and deep learning, achieving a correlation of 0.569 using doc2vec. Building on this, a Siamese BiLSTM model was employed in [17], demonstrating significant improvements in essay grading with the ASAP dataset. Similarly, in [18], a hybrid approach combined a Siamese network for sentence similarity that further expanded the applicability of deep learning in grading tasks. These studies collectively underscore the evolution of embedding techniques and model architecture in ASAG, paving the way for further research into language-specific solutions and enhanced explainability.

In [19], MMR and GAN-LCS methods were used, achieving 91% accuracy in handling diverse responses. In [20], BERT outperformed other models in short-answer scoring using the QWK metric. In [21], attention mechanisms were

integrated with bidirectional RNNs, improving performance by 10%. D-DAS [22] achieved up to 89% accuracy in descriptive response grading with bidirectional LSTM. In [23], domain-general and specific information were combined using CBOW and LSTM for semi-open-ended questions. In [24], a stacking model used XGBoost and SMOTE for class imbalance, achieving an F1-score of 0.821. In [25], MaLSTM was employed with sense vectors for ASAG, demonstrating promising results. In [26], GWO was used to optimize LSTM hyperparameters, improving accuracy. In [27], LLMs (e.g., GPT-4, Gemini) were tested on Arabic GAT questions, finding that they handle complex Arabic reasoning fairly well. In [28], few-shot prompting was utilized with LLMs for short answer grading, demonstrating that large models such as GPT-4 can achieve close to human grading quality. In [29], the ASAG 2024 benchmark was proposed, combining multiple datasets to improve model evaluation and generalization across diverse question types. In [30], it was demonstrated that AraBERT performed effectively on Arabic educational texts, supporting its use in the automatic grading of Arabic short answers.

B. Explainability for ASAG

Explainability is essential in ASAG to ensure that educators and students understand the grading results. In [31], XAI methods were investigated, showing that the display of predicted scores with model-student answer comparisons improves transparency and trustworthiness. In [32], XAI techniques were evaluated in automated essay scoring using the predictive decision rule framework and various SHAP methods. This study demonstrated the effectiveness of TreeSHAP in improving descriptive accuracy. The method in [33] used LLMs to generate detailed feedback using the EngSAF dataset, focusing on formative assessment rather than simple scores. In [34], a self-explainable rationale-driven multi-trait essay scoring framework was proposed, which generated a numerical score alongside a human-interpretable rationale, enhancing trust in automated decisions.

C. Summary of Findings

As shown in Table I, recent advances in ASAG focus primarily on improving grading accuracy through deep learning models and embedding techniques. Techniques such as BERT, LSTM, and attention mechanisms have demonstrated substantial improvements in grading performance, especially when combined with advanced embeddings. However, a significant research gap remains, as although there has been considerable progress in enhancing accuracy, benchmarking these models alongside various embedding techniques is underexplored. Moreover, most current ASAG models do not adequately address the issue of explainability. Systems such as those proposed in [31, 32] have made strides toward integrating explainability, but mainly focus on non-Arabic languages and lack cultural and linguistic adaptation for Arabic language contexts.

This gap underscores the need for comprehensive research that balances performance and interpretability. Most existing models excel in one of these aspects, either achieving high accuracy or offering explanations, but rarely do both. This work aimed to address this dual challenge by developing an ASAG system that not only ensures accurate grading but also integrates robust explainability features, offering meaningful and transparent feedback. In the implementation of the proposed ASAG system, Transformer-Interpret was selected based on its demonstrated effectiveness in domains that demand both high predictive accuracy and interpretability, such as clinical outcome analysis and mental health research [35, 36]. The proposed ASAG system aims to achieve a balanced approach, providing not just accurate assessments but also insightful feedback that can be trusted by both students and educators.

TABLE I. COMPARISON OF RECENT ASAG APPROACHES

Ref	Model	Language	Dataset	Explainability	Best performance
[16]	word2vec, Glove, fasttext, Elmo, doc2vec, InferSent and skip-Thought	English	ASAGV2.0	NO	doc2vec
[17]	SBLSTMA	English	ASAP	NO	SBLSTMA
[20]	CharCNN, CNN, bi-LSTM, and BERT	English	ASAP	No	BERT
[22]	Simple LSTM, Deep LSTM, Bi-LSTM	English	D-DAS dataset	No	BiLSTM
[26]	LSTM with GWO	Arabic	Collected	NO	LSTM with GWO
[31]	MLP neural network with SHAP	English	ASAP	Yes	MLP with SHAP

III. METHODOLOGY

This study proposes an automatic grading system for Arabic short-answer questions by leveraging advanced word and sentence embedding techniques, including BERT models, to capture the semantic nuances of Arabic text, and employing attention-enhanced neural networks. Additionally, Transformer-Interpret is employed to generate interpretable feedback for students and educators, thereby enhancing the transparency and trustworthiness of the system.

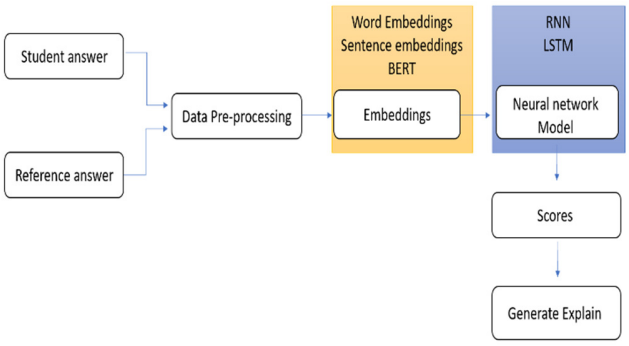

Fig. 1. The proposed system.

As illustrated in Figure 1, the method consists of four main steps: (i) Data preprocessing, where student and reference responses are standardized and cleaned; (ii) Embedding generation, which uses techniques such as word embeddings, sentence embeddings, and BERT to encode semantic meaning; (ii) Neural network modeling, where RNN and LSTM models

are enhanced with an attention mechanism to predict answer grades; and (iv) Explainability generation, using Transformer-Interpret to create clear and interpretable feedback for the assigned scores.

A. Datasets

This study tests the performance of the models on three distinct datasets. Table II provides an overview of these datasets.

TABLE II. OVERVIEW OF THE DATASETS UTILIZED

Dataset	Language	Course	Questions	Answers
AR-ASAG	Arabic	Cybercrime	48	2133
AT-ASAG	Arabic	Biology	15	9842
Dataset 3 (combined)	Arabic	Cybercrime and Biology	63	11,975

1) AR-ASAG-Dataset

The AR-ASAG dataset [37] is an open-source dataset designed for the automatic grading of short answers in Arabic. This dataset, available in multiple formats, was compiled from three different exams on cybercrime courses, involving responses from three cohorts of MSc students. It includes 48 questions with 2133 pairs of model and student answers. Table III provides a sample of the AR-ASAG dataset. The questions in this dataset cover five distinct types:

- "عرف ": Define
- "اشرح": Explain
- "ما النتائج المترتبة على": What consequences…?
- "علل": Justify
- "ما الفرق": What is the difference?

TABLE III. SAMPLE OF AR-ASAG-DATASET

Sample question	عرف مصطلح أمن المعلومات؟	Mark 1	Mark 2	AVG
Model answer	حماية وتأمين كافة الموارد المستخدمة في معالجة المعلومات من منشآت نفسها والأفراد العاملين فيها وأجهزة الحاسب المستخدمة فيها ووسائط المعلومات التي تحتوي على البيانات وذلك في جميع مراحل تواجد المعلومة (التخزين – النقل –المعالجة)			
Student answer 1	هي الوسائل والأدوات والسياسات والإجراءات المستخدمة لضمان خصوصية وتكامل وتوافر المعلومات ومنع الوصول إليها أو حذفها أو مسحها بدون حق.	2.4	2.4	2.4
Student answer 2	هي الوسائل التقنية والإدارية التي يجب توفيرها لحماية معالجة المعلومات (تخزينها، نقلها، معالجتها).	1.8	2.1	1.95
Student answer 3	هو تأمين المنشأة، الموظفين، الأنظمة، الأجهزة، ووسائط الإعلام التي تحتوي على المعلومات في جميع مراحل المعلومات (النقل - المعالجة).	2.4	2.1	2.25

Two teachers independently evaluated the students' responses, assigning scores on a scale from 0 (completely incorrect) to 5 (completely correct). The average of these scores was taken as the gold standard. The range of AR-ASAG scores was changed from 0 to 3 using the min-max scaling method, which is a common way to normalize the data to a certain range. This ensured that the scores were consistent and made it easier to combine them with the other datasets. The equation for this transformation is:

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (1)$$

where X represents the original scores. Figure 2 illustrates the distribution of scores.

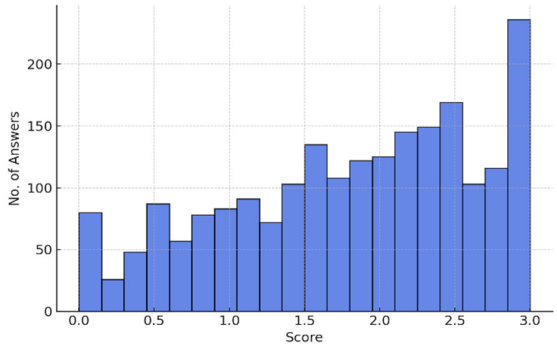

Fig. 2. Scores distribution in AR-ASAG-Dataset.

2) AT-ASAG-Dataset

The AT-ASAG dataset was created by translating the PT-ASAG [38] dataset into Arabic. Experts meticulously performed this translation process and regraded the student responses to correct any grading discrepancies. The dataset contains 15 questions related to human biology, contributed by five elementary school teachers from the Pampa Federal University, Brazil. A total of 659 students, including both elementary and high school students, took the test, and 14 undergraduate biology students in their final year evaluated the responses. Table IV provides a sample of the translated and re-evaluated questions and answers, alongside the original scores.

TABLE IV. SAMPLE OF AT-ASAG-DATASETS

QID	Model Answer	Student Answer	Mark	Old mark
36	النباتات ، تحتوي الخلية على جدار خلوي يغطيها خارجيًا. يتكون الجدار الخلوي بشكل أساسي من السيليلوز، و للخلايا الحيوانية والنباتية شكل مختلف فالخلية الحيوانية لها شكل غير منتظم ، بينما الخلية النباتية لها شكل ثابت.	للحيوان نواة وليس النبات	0	0
37	غشاء البلازما: الذي ينظم تبادل المواد بين الخلية والبيئة الخارجي السيتوبلازم : مادة هلامية داخلية يتم دمج العديد من العضيات الأصغر فيها النواة: بنية مرتبطة بتنسيق وظائف الخلية وتكاثر الخلايا. الحمض النووي موجود في النواة.	غشاء الخلية ، السيتوبلازم ، الغشاء النووي ، النواة	1	1
38	يتميز النسيج الظهاري بالخصائص التالية: نقص المساحة بين الخلايا ، ونقص الأوعية الدموية والقدرة الممتازة على تجديد الخلايا.	الخلايا معًا (متجاورة) ، لا تحتوي على أوعية دموية	2	1
41	إنها الطريقة التي يعمل بها انتقال الصفات الوراثية من خلال الجينات التي تنتقل من الآباء إلى الأبناء.	هذه هي الخصائص التي تنتقل من الوالد إلى الطفل من خلال الجينات.	3	2

Grading involved a scale of 0 to 3, where 0 represents an entirely incorrect answer, and 3 represents a fully correct answer. Due to the limited availability of large-scale, publicly accessible Arabic short answer grading datasets, this dataset was translated, identifying AR-ASAG as the only other such dataset. Figure 3 shows the distribution of the scores.

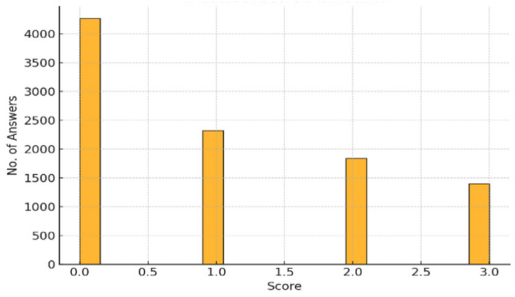

Fig. 3. Scores distribution in AT-ASAG dataset.

3) Dataset 3

This dataset was created by integrating the corpus of both AR-ASAG and AT-ASAG datasets through systematic merging, resulting in a comprehensive benchmark comprising 63 distinct questions and 11,975 student responses. This unified resource enables evaluating models' performance across disparate knowledge domains and the handling of both native and translated Arabic datasets. The merging protocol ensured the preservation of the original question structures, scoring rubrics, responses' originality, linguistic characteristics, and domain-specific evaluation metrics. Figure 4 illustrates the distribution of grades regarding the number of student responses.

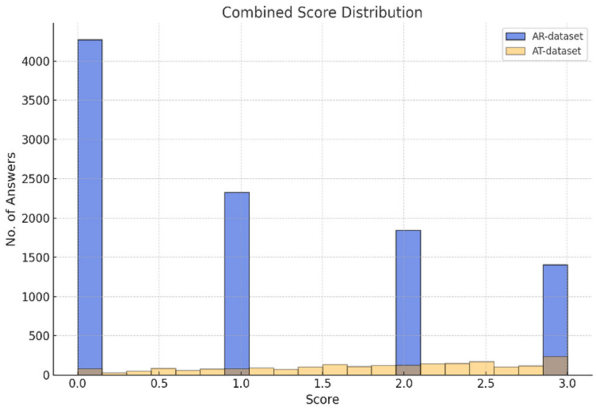

Fig. 4. Grade distribution across student responses in dataset 3.

B. Data Preprocessing

Preprocessing is a crucial step in ensuring the quality and reliability of the model's predictions. Initially, the raw text data was normalized to standardize Arabic script, including the removal of diacritics and the normalization of various forms of letters. Stop words and punctuation were also removed to reduce noise. Then, tokenization was applied to separate the text into individual words or tokens. Infrequent terms were filtered out to improve model generalization, and the remaining tokens were converted into vectors using pre-trained word embeddings. This rigorous preprocessing pipeline helped to refine the dataset, making it suitable for effective training and evaluation of the models.

C. Word Embedding Models

GloVe and FastText embeddings were used with RNN and LSTM architectures. Rare words were removed after tokenization to improve generalization. The embeddings were then processed through recurrent layers, followed by linear transformations and dropout to enhance the robustness of the model.

D. Sentence Embedding Models

The Universal Sentence Encoder and Arabic BERT (aubmindlab/bert-base-arabertv01) were employed for sentence-level representations. These embeddings were passed through RNN/LSTM layers. Finally, the output was processed through linear and ReLU-activated layers for dimensionality reduction and feature learning.

E. AraBERT

AraBERT [39] is a pre-trained Arabic language model based on BERT, trained on 3 billion words from sources such as Arabic Wikipedia. It comes in two versions: BERT Base (12 layers, 110 million parameters) and BERT Large (24 layers, 330 million parameters). Although BERT Base has a simpler architecture than BERT Large, it always performed the best in our tests, finding the best balance between being fast and being able to understand the complicated Arabic language. For input preparation, special tokens were incorporated, such as [CLS] for classification tasks and [SEP] to demarcate sentence boundaries. A sequence is arranged as follows: [CLS] + tokenized (student answer) + [SEP] + tokenized (model answer) + [SEP]. Padding was added to maintain uniform sequence length and facilitate batch processing. Attention masks were generated to distinguish between actual tokens and padding, and then the preprocessed input was fed into the BERT encoder for further processing.

F. Recurrent Neural Networks (RNN)

RNNs process sequences by keeping a hidden state that preserves information from earlier points in the data. This makes them effective for modeling temporal patterns in language data. RNNs help capture the contextual flow between words and sentences, improving the understanding of Arabic short answers.

G. Long Short-Term Memory Networks (LSTM)

LSTM networks, an advanced form of RNNs, are designed to capture long-term dependencies in sequences using specialized gates to control information flow. In this task, LSTMs help preserve important contextual information across Arabic sentences, enhancing the modeling of student answers.

H. Attention Mechanism Integration

An attention mechanism was incorporated to capture the most salient parts of the student and the model responses. Specifically, after extracting contextual embeddings from the recurrent layer, a scaled dot-product attention layer was applied to learn which tokens in the student response should contribute more heavily to the final grade prediction. This module assigns higher weights to tokens that align more closely with the model answer, thus amplifying relevant portions of the input while diminishing the impact of less informative tokens.

I. Implementation Details

- Recurrent encoder: The tokenized student and model answers are fed into the RNN or LSTM encoder to capture sequential dependencies.
- Attention layer: The hidden states from the encoder are passed through a scaled dot-product attention mechanism. Each token obtains an attention weight that reflects its importance in matching the reference answer.
- Context vector: The encoder outputs are aggregated and weighted by the attention scores to produce a context vector representing the student's response focus.
- Classification/Regression head: This context vector is then fed into a dense layer to predict the final grading score.

J. Explainability of the Model

Model explainability is a crucial aspect of machine learning and deep learning models. To comprehend the functioning of so-called black box models and gain insight into how they generate specific predictions or grades, systems should provide feedback to students, explaining the assigned scores. This transparency is essential for users to trust the results produced by ASAG models. This study utilized Transformers-Interpret, an extension of the Captum library, to enhance the interpretability of the model. The Transformer-Interpret Python package leverages Integrated Gradients [40], along with its variant, Layer Integrated Gradients. Integrated Gradients is a vital technique for understanding the significance of input features in model predictions. It computes the gradient of the model's output regarding its input along a trajectory from a designated baseline to the target input and integrates these gradients along the path. This approach enhances interpretability and transparency by revealing how each input feature contributes to the model's output. In addition, it incorporates principles such as the implementation invariance of gradients and algorithmic sensitivity, further improving its analytical capabilities.

K. Experiment

First, two different word embedding models, GloVe and FastText, were used in two separate instances: once paired with an RNN model and once with an LSTM model. Initially, the answers (student responses and reference answers) were tokenized into sequences of single words, each referred to as a token. Infrequent words were removed to reduce the vocabulary size, which improves the model's generalization by focusing on more prevalent and informative terms. This approach also addresses data sparsity challenges, resulting in a model that is easier to manage and more efficient. Next, a pre-trained file was loaded, containing a dictionary of words with their corresponding word vectors. The words in the responses were matched to the entries in the dictionary, producing a set of word vectors for the recognized terms. An RNN model, consisting of three RNN layers, processes the input through recurrent connections, a linear layer for regression, and a dropout layer. The same steps were used for the LSTM model, which has two LSTM layers: an output that maps the information gathered by the previous layers to a single dimension and a dropout that makes the model more general.

Second, two sentence embedding approaches were implemented: TensorFlow Hub's Universal Sentence Encoder and Sentence Transformers with arabertv01 (a pre-trained Arabic BERT model). Both embeddings were processed through separate RNN and LSTM architectures. The pipeline began by tokenizing the student and reference responses into discrete tokens. These tokenized inputs were then transformed into dense vector representations using the selected embedding models. The RNN architecture employed two parallelized RNN layers, each configured with a single hidden layer. After the input sequences are processed by each branch separately, there is a linear layer, a dropout layer, and a ReLU activation function that adds non-linearity and lets the model learn complex patterns. For the LSTM model, two separate LSTM branches handle the paired input sequences, followed by a linear layer, a dropout layer, and a ReLU activation function.

By explicitly modeling which tokens are most relevant through attention, this approach can enhance grading accuracy and provide interpretable feedback to students and instructors. The attention weights can be visualized, allowing stakeholders to understand why certain portions of an answer were deemed critical for the assigned grade. This added transparency supports the goal of developing a trustworthy automated scoring system for Arabic short-answer questions.

The grid search method was used to find the best values for the number of epochs, batch size, initial learning rate, and other hyperparameters while building the ASAG model. The following sets of values were explored.

- Learning rate: ranges from 1e-4 to 8e-4, with a step size of 1e-4.
- Dropout rates: Ranges from 0.1 to 0.5, with a step size of 0.1.
- Number of epochs: Ranges from 500 to 3000, with a step size of 500.
- Hidden layer dimensions: Ranges from 100 to 700, with a step size of 50.
- Number of hidden layers: Ranges from 1 to 4, with a step size of 1.
- Batch size: Ranges from 1 to 64, with increments of 1, 16, 32, and 64.

Table V describes the hyperparameters for the RNN model.

TABLE V. PARAMETER VALUES FOR THE RNN MODEL

RNN	Sentence Transformers	Universal Sentence Encoder	FastText	GloVe
Learning rate	1e-4	1e-4	5e-4	1e-4
Embedding size	768	512	300	300
Number of hidden layers	2	2	1	1
Hidden layer dimension	350	350	356	500
Dropout	0.3	0.3	0.1	0.1
Number of epochs	3000	3000	2000	1000
Batch size	1	1	1	1

For the LSTM model, similar hyperparameters were explored, as shown in Table VI.

TABLE VI. PARAMETER VALUES FOR THE LSTM MODEL

LSTM	Sentence transformers	Universal Sentence Encoder	FastText	GloVe
Learning rate	1e-4	1e-4	5e-3	1e-4
Embedding size	768	512	300	300
Number of hidden layers	2	2	1	1
Hidden layer dimension	350	256	1000	100
Dropout	0.3	0.3	0.1	0.2
Number of epochs	3000	2000	2000	2000
Batch size	1	1	1	1

Finally, three distinct regression models were developed, all based on the AraBERT architecture. The inputs were tokenized using Byte-Pair Encoding [8], which effectively handles unseen terms by decoding words such as "sleeping" as [sleep + ##ing]. Special tokens, such as [CLS] for classification tasks and [SEP] to differentiate between student and reference answers, were added, mapping these tokens to their corresponding IDs. A maximum input length of 128 was established, and zeros were used to pad or truncate shorter inputs appropriately. Attention masks were used to distinguish real from padding tokens.

The first model is based on the AraBERT architecture, with all layers derived from the PyTorch library. The AraBERT embedding layer captures the rich linguistic context of Arabic text, followed by a dropout layer to enhance model robustness and a fully connected layer to map the pooled features to a regression output. The second model consists of the AraBERT embedding layer, a dropout layer, an RNN layer to capture sequential dependencies, and an attention layer to highlight the most relevant features from the RNN outputs. This ensures that the model effectively understands the order and context of words within the text. The third model similarly integrates the AraBERT embedding layer, a dropout layer, an LSTM layer for capturing temporal relationships, followed by an attention layer to focus on critical parts of the LSTM output. In the context of ASAG, the LSTM processes AraBERT-based embeddings step-by-step, while the attention mechanism enhances the model's ability to identify and emphasize key aspects of short answers' context and structure. In both cases, a fully connected layer follows for the final prediction.

A grid search strategy was employed to determine the most suitable hyperparameter values, exploring the following ranges:

- Learning rate: ranges from 1e-6 to 1e-4, with a step size of 9e-6.
- Number of hidden layers: Ranges from 1 to 4, with a step size of 1.
- Hidden layer dimensions: Ranges from 64 to 512, with a step size of 64.
- Number of epochs: Ranges from 10 to 30, with a step size of 5.

Table VII shows the hyperparameters for these models.

TABLE VII. PARAMETERS FOR BERT WITH LSTM AND RNN

Model	BERT+LSTM	BERT+RNN
Learning rate	1e-5	1e-5
Number of hidden layers	1	1
Hidden layer dimension	256	256
Number of epochs	15	15

Google Colab was used for the experiments, offering efficient training and testing of deep learning models without relying on specialized local hardware. There are hidden layers that range from 1 to 4, each with a step size of 1, to build and train neural networks. A uniform approach was maintained to testing and validation across all experiments. The dataset was divided into 80% training, 10% for validation, and the remaining 10% for testing. The AdamW optimizer was used with learning rates and batch sizes as shown in Tables V-VII.

IV. RESULT AND DISCUSSION

A. Evaluation Measures

The performance of the proposed ASAG model was evaluated using RMSE and PCC. RMSE quantifies the average magnitude of differences between predicted and actual values. It is derived by taking the square root of the mean squared differences, as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad (2)$$

where N denotes the number of samples, y_i refers to the actual value, and $\hat{y}_i$ refers to the predicted value. Lower RMSE values reflect stronger model accuracy, with a value of 0 representing a perfect match between predictions and actual outcomes.

PCC assesses the strength and direction of the linear relationship between predicted and true values.

$$PCC = \frac{\sum_{i=1}^{N}(y_i-\bar{y}_i)(\hat{y}_i-\bar{\hat{y}})}{\sqrt{\sum_{i=1}^{N}(y_i-\bar{y})^2\sum_{i=1}^{N}(\hat{y}_i-\bar{\hat{y}})^2}} \quad (3)$$

where y_i is the actual value, $\hat{y}_i$ is the predicted value, $\bar{y}$ represents the mean of actual values, and $\bar{\hat{y}}$ is the mean of predicted values. The result always has a value between -1 and 1, where a value of correlation close to 1 indicates a strong positive linear relationship between predicted and actual values, while a value near -1 signifies a strong negative linear relationship. A coefficient close to 0 indicates no linear relationship or a weak one.

B. Experimental Results

Tables VIII-X present and compare the results of the proposed ASAG system on the test data for the three datasets.

C. Discussion

The evaluation of various models across the three datasets offers valuable insights into the impact of model architecture, embedding techniques, and dataset characteristics on the ASAG systems in Arabic.

TABLE VIII. RESULT FOR THE RNN MODEL

RNN	Dataset 1		Dataset 2		Dataset 3	
	PCC	RMSE	PCC	RMSE	PCC	RMSE
Sentence embedding using transformer	0.597	0.939	**0.91**	0.267	0.896	0.307
universal-sentence-encoder	0.554	0.964	0.834	0.445	0.809	0.463
fastest	0.219	1.192	0.709	0.577	0.774	0.457
glove	0.320	1.169	0.737	0.446	0.674	0.533
BERT+RNN	**0.813**	**0.652**	0.906	**0.1828**	**0.914**	**0.212**

TABLE IX. RESULTS FOR THE LSTM MODEL

LSTM	Dataset 1		Dataset 2		Dataset 3	
	PCC	RMSE	PCC	RMSE	PCC	RMSE
Sentence embedding using transformer	0.619	0.913	**0.920**	0.252	0.896	0.299
universal-sentence-encoder	0.632	0.884	0.908	0.288	0.888	0.329
Fastest	0.338	1.122	0.884	0.221	0.871	0.270
Glove	0.576	0.960	0.858	0.273	0.819	0.334
BERT+LSTM	**0.803**	**0.679**	0.907	**0.1827**	**0.932**	**0.182**

TABLE X. RESULTS FOR THE BERT MODEL

Model	Dataset 1		Dataset 2		Dataset 3	
	PCC	RMSE	PCC	RMSE	PCC	RMSE
BERT	0.811	0.688	0.921	0.201	0.923	0.206

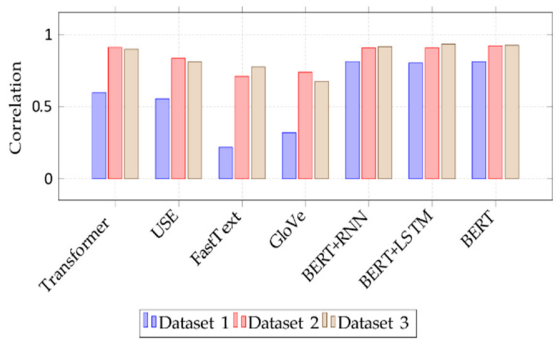

Fig. 5. Correlation comparison across models and datasets.

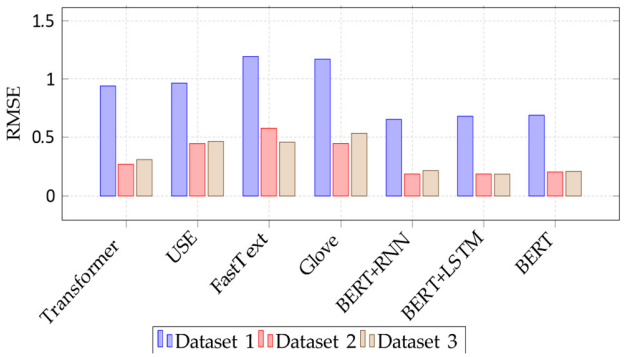

Fig. 6. RMSE comparison across models and datasets.

As shown in Figures 5 and 6, the BERT-based models (standalone BERT, BERT+RNN, and BERT+LSTM) consistently outperformed models relying on other embeddings, indicating that BERT's contextual embeddings are critical for accurate short answer grading. The results also show that, in general, LSTM outperformed RNN, as they generally achieved higher correlations and lower RMSE values, particularly on Datasets 2 and 3. This is likely due to LSTM's ability to retain long-term dependencies and better handling of sequential data, which is crucial for understanding the structure of student responses.

Standalone BERT achieved the highest correlation on AT-ASAG (0.921) and combined data (0.923), outperforming hybrid models (BERT+RNN/LSTM) in these settings. This may be attributed to its self-attention mechanism, which captures semantic relationships without needing sequential modeling. However, BERT+LSTM marginally surpassed standalone BERT on AR-ASAG (0.803 vs. 0.811) and the combined dataset (0.932 vs. 0.923), indicating that LSTMs can enhance BERT's performance for native Arabic answers.

Contextual embeddings represented by BERT-based models (standalone or hybrid) rank first across all datasets, confirming that contextual embeddings are essential for Arabic ASAG. Traditional embeddings (FastText and GloVe) perform poorly on AR-ASAG (PCC≤0.338 for LSTM), but improve on AT-ASAG (FastText PCC=0.884 with LSTM). This highlights their reliance on simpler, translated text structures and inadequacy for native Arabic's linguistic nuances. Universal Sentence Encoder (USE) bridges this gap moderately (e.g., LSTM+USE PCC is 0.632 on AR-ASAG vs. 0.908 on AT-ASAG), but remains inferior to BERT.

Figure 7 illustrates the positive correlation between the size of the dataset and the performance of the model. As the volume of training data increases, both the correlation coefficient and the RMSE improve. This trend is consistent across different embeddings and architectures, highlighting the importance of extensive and high-quality data for robust model training. The results show that Dataset 1 (AR-ASAG) consistently yielded the lowest performance across all models. Even the best-performing model (BERT+LSTM) achieved a correlation of only 0.803. However, Dataset 2 (AT-ASAG) showed significantly better results, with all models achieving higher correlations and lower RMSEs. This improvement is likely because manually translated answers (such as AT-ASAG) often exhibit more consistent and simplified sentence structures, whereas native responses (such as AR-ASAG) introduce real-world noise that challenges models [41, 42]. Additionally, the AR-ASAG dataset suffers from limited scope, size, linguistic variability, and coverage, which restricts the model's ability to generalize, especially when compared to more diverse or translated datasets. This directly explains the observed performance gap. Finally, Dataset 3 (combined) consistently produced the best results, especially for BERT-based models. This demonstrates the positive impact of increasing the size and diversity of the dataset, which enhances the generalization of the model and reduces overfitting. The combination of native and translated data introduces a broader range of vocabulary, sentence structures, and answer styles, allowing models to learn more comprehensive scoring patterns.

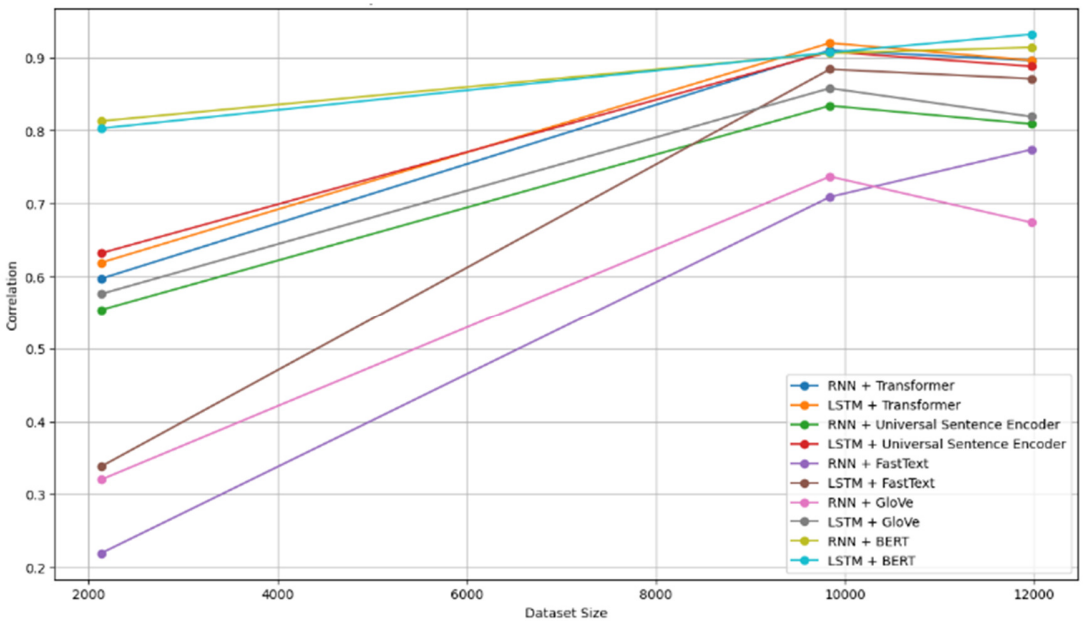

Fig. 7. Impact of dataset Size on model performance.

D. Explainability

This study also investigated the use of Transformers-Interpret to visualize the average attention across all layers of the model for two inputs: the student's response and the reference answer. This visualization assigns an attention score to each token or sub-token in the input text, indicating its importance in the model's prediction. Positive attention scores signify tokens that positively influence the predicted outcome, while negative scores indicate a detrimental impact. For instance, in Figure 8, tokens like الاجسام المضادة (antibodies) exhibit notably high positive attention scores, highlighting their significance in the prediction. In contrast, tokens such as الدم (blood) and الحمراء (red) possess negative attention scores, which implies a potentially adverse effect on prediction. These attention scores offer valuable insights into how different parts of the input text contribute to the model's decision-making process, enhancing interpretation and understanding of the model's behavior.

```
('0.2283732943086539 ,'الا'),
('##0.05904757192030185 ,'ج'),
('##0.19960241131721318 ,'سام'),
('0.5465997470996796 ,'المضادة'),
('0.1003934366902351- ,'في'),
('0.07195651457220972- ,'خلايا'),
('0.48433823017547994- ,'الدم'),
('0.2208152420001942- ,'الحمراء'),
```

Fig. 8. Token-level attribution visualization.

Subsequently, Figure 9 visualizes the attribution scores using a heatmap across lexical units. Regions highlighted in red signify a negative contribution to the grading, while those in green denote a positive influence. White regions indicate neutrality or no discernible contribution.

Fig. 9. Interpretation visualization.

V. CONCLUSION

This study addresses the growing need for efficient and transparent assessment of Arabic short answers, recognizing the limitations of manual grading, such as time consumption, subjectivity, and the lack of immediate feedback. Although prior ASAG research has made significant progress in improving grading accuracy, a notable gap remains in achieving both high performance and strong explainability, particularly for Arabic language contexts. This study developed an Arabic ASAG system that compares various word- and sentence-level embedding techniques within deep learning models, including RNN, LSTM, and BERT architectures. The models were evaluated on three Arabic datasets of different sizes and types, using PCC and RMSE. One of these datasets was newly created by translating an existing Portuguese dataset into Arabic, with experts translating and regrading the student

responses to correct for any grading discrepancies, ensuring the reliability and relevance of the dataset for Arabic ASAG tasks.

The results show that model performance increases proportionally with the size of the training set, a trend observed across various model structures and embeddings. LSTM networks consistently outperformed RNNs, demonstrating their superior ability to learn long-range dependencies in text data. The best performance metrics across all datasets were achieved by a hybrid architecture using BERT embeddings with LSTM (BERT+LSTM), highlighting the strength of contextualized embeddings in regression-based ASAG tasks. A key contribution of this work is the integration of Transformer-Interpret, which enables the system to provide transparent and comprehensible feedback along with accurate grading. This dual focus on performance and interpretability distinguishes the proposed ASAG framework and represents a significant step toward more trustworthy and effective automated assessment in Arabic educational settings.

Future endeavors will aim to acquire a large and high-quality Arabic dataset and explore different deep learning architectures, such as CNNs and transformer-based models, such as GPT, to further optimize the ASAG task. Additionally, alternative interpretability methods, beyond Transformer-Interpret, should be investigated, such as model-agnostic techniques, such as LIME and SHAP, to provide insights into predictions regardless of model architecture.

REFERENCES

[1] L. Yuan and S. J. Powell, MOOCs and open education: Implications for higher education. JISC cetis, 2013.

[2] E. Badger and B. Thomas, "Open-Ended Questions in Reading," *Practical Assessment, Research, and Evaluation*, vol. 3, no. 1, Jan. 1991, https://doi.org/10.7275/fryf-z044.

[3] R. E. Bennett, "On the Meanings of Constructed Response," *ETS Research Report Series*, vol. 1991, no. 2, 1991, https://doi.org/10.1002/j.2333-8504.1991.tb01429.x.

[4] E. B. Page, "The Imminence of... Grading Essays by Computer," *The Phi Delta Kappan*, vol. 47, no. 5, pp. 238–243, 1966.

[5] P. W. Foltz, D. Laham, and T. K. Landauer, "The Intelligent Essay Assessor: Applications to Educational Technology," *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, vol. 1, no. 2, 1999.

[6] A. Adamson, A. Lamb, and R. M. December, "Automated essay grading," in *Proceedings of the Conference on Artificial Intelligence in Education*, Québec, Canada, 2014, pp. 27–31.

[7] B. Riordan, A. Horbach, A. Cahill, T. Zesch, and C. M. Lee, "Investigating neural architectures for short answer scoring," in *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, Copenhagen, Denmark, Jun. 2017, pp. 159–168, https://doi.org/10.18653/v1/W17-5017.

[8] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, MN, USA, Mar. 2019, pp. 4171–4186, https://doi.org/10.18653/v1/N19-1423.

[9] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[10] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching Word Vectors with Subword Information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, Jun. 2017, https://doi.org/10.1162/tacl_a_00051.

[11] D. Cer *et al.*, "Universal Sentence Encoder for English," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Brussels, Belgium, Aug. 2018, pp. 169–174, https://doi.org/10.18653/v1/D18-2029.

[12] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks." arXiv, Aug. 27, 2019, https://doi.org/10.48550/arXiv.1908.10084.

[13] V. Belle and I. Papantonis, "Principles and Practice of Explainable Machine Learning," *Frontiers in Big Data*, vol. 4, Jul. 2021, https://doi.org/10.3389/fdata.2021.688969.

[14] A. B. Arrieta *et al.*, "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, Jun. 2020, https://doi.org/10.1016/j.inffus.2019.12.012.

[15] S. Burrows, I. Gurevych, and B. Stein, "The Eras and Trends of Automatic Short Answer Grading," *International Journal of Artificial Intelligence in Education*, vol. 25, no. 1, pp. 60–117, Mar. 2015, https://doi.org/10.1007/s40593-014-0026-8.

[16] S. Hassan, A. A. Fahmy, and M. El-Ramly, "Automatic short answer scoring based on paragraph embeddings," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 10, pp. 397–402, 2018.

[17] G. Liang, B. W. On, D. Jeong, H. C. Kim, and G. S. Choi, "Automated Essay Scoring: A Siamese Bidirectional LSTM Neural Network Architecture," *Symmetry*, vol. 10, no. 12, Dec. 2018, Art. no. 682, https://doi.org/10.3390/sym10120682.

[18] P. Patil and A. Agrawal, "Auto Grader for Short Answer Questions," presented at the CS229: Machine Learning, Stanford University, 2018.

[19] F. S. Pribadi, A. E. Permanasari, and T. B. Adji, "Short answer scoring system using automatic reference answer generation and geometric average normalized-longest common subsequence (GAN-LCS)," *Education and Information Technologies*, vol. 23, no. 6, pp. 2855–2866, Nov. 2018, https://doi.org/10.1007/s10639-018-9745-z.

[20] K. Surya, E. Gayakwad, and M. Nallakaruppan, "Deep learning for short answer scoring," *International Journal of Recent Technology and Engineering*, vol. 7, no. 6, pp. 1712–1715, 2019.

[21] T. Gong and X. Yao, "An attention-based deep model for automatic short answer score," *International Journal of Computer Science and Software Engineering*, vol. 8, no. 6, pp. 127–132, 2019.

[22] N. George, P. J. Sijimol, and S. M. Varghese, "Grading descriptive answer scripts using deep learning," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 5, 2019.

[23] L. Zhang, Y. Huang, X. Yang, S. Yu, and F. Zhuang, "An automatic short-answer grading model for semi-open-ended questions," *Interactive Learning Environments*, vol. 30, no. 1, pp. 177–190, Jan. 2022, https://doi.org/10.1080/10494820.2019.1648300.

[24] R. A. Rajagede and R. P. Hastuti, "Stacking Neural Network Models for Automatic Short Answer Scoring," *IOP Conference Series: Materials Science and Engineering*, vol. 1077, no. 1, Oct. 2021, Art. no. 012013, https://doi.org/10.1088/1757-899X/1077/1/012013.

[25] C. N. Tulu, O. Ozkaya, and U. Orhan, "Automatic Short Answer Grading With SemSpace Sense Vectors and MaLSTM," *IEEE Access*, vol. 9, pp. 19270–19280, 2021, https://doi.org/10.1109/ACCESS.2021.3054346.

[26] M. A. Salam, M. A. El-Fatah, and N. F. Hassan, "Automatic grading for Arabic short answer questions using optimized deep learning model," *PLOS ONE*, vol. 17, no. 8, 2022, Art. no. e0272269, https://doi.org/10.1371/journal.pone.0272269.

[27] M. D. Alahmadi, M. Alharbi, A. Tayeb, and M. Alshangiti, "Evaluating Large Language Models' Proficiency in Answering Arabic GAT Exam Questions," *Engineering, Technology & Applied Science Research*, vol. 14, no. 6, pp. 17774–17780, Dec. 2024, https://doi.org/10.48084/etasr.8481.

[28] C. Zhao, M. Silva, and S. Poulsen, "Language Models are Few-Shot Graders." arXiv, Feb. 18, 2025, https://doi.org/10.48550/arXiv.2502.13337.

[29] G. Meyer, P. Breuer, and J. Fürst, "ASAG2024: A Combined Benchmark for Short Answer Grading," in *Proceedings of the 2024 on ACM Virtual Global Computing Education Conference*, Sep. 2024, pp. 322–323, https://doi.org/10.1145/3649409.3691083.

[30] H. Alamoudi *et al.*, "Arabic Sentiment Analysis for Student Evaluation using Machine Learning and the AraBERT Transformer," *Engineering, Technology & Applied Science Research*, vol. 13, no. 5, pp. 11945–11952, Oct. 2023, https://doi.org/10.48084/etasr.6347.

[31] T. Schlippe, Q. Stierstorfer, M. ten Koppel, and P. Libbrecht, "Explainability in Automatic Short Answer Grading," in *Artificial Intelligence in Education Technologies: New Development and Innovative Practices*, Singapore, 2023, pp. 69–87, https://doi.org/10.1007/978-981-19-8040-4_5.

[32] V. Kumar and D. Boulanger, "Explainable Automated Essay Scoring: Deep Learning Really Has Pedagogical Value," *Frontiers in Education*, vol. 5, Oct. 2020, https://doi.org/10.3389/feduc.2020.572367.

[33] D. Aggarwal, P. Sil, B. Raman, and P. Bhattacharyya, "'I understand why I got this grade': Automatic Short Answer Grading with Feedback." arXiv, Jun. 23, 2025, https://doi.org/10.48550/arXiv.2407.12818.

[34] H. Do, S. Ryu, and G. G. Lee, "Teach-to-Reason with Scoring: Self-Explainable Rationale-Driven Multi-Trait Essay Scoring." arXiv, Feb. 28, 2025, https://doi.org/10.48550/arXiv.2502.20748.

[35] Y. Zhu *et al.*, "Using natural language processing on free-text clinical notes to identify patients with long-term COVID effects," in *Proceedings of the 13th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, May 2022, pp. 1–9, https://doi.org/10.1145/3535508.3545555.

[36] V. M. Nguyen *et al.*, "Conceptualizing Suicidal Behavior: Utilizing Explanations of Predicted Outcomes to Analyze Longitudinal Social Media Data," in *2023 International Conference on Machine Learning and Applications (ICMLA)*, Jacksonville, FL, USA, Dec. 2023, pp. 2095–2102, https://doi.org/10.1109/ICMLA58977.2023.00316.

[37] L. Ouahrani and D. Bennouar, "AR-ASAG An ARabic Dataset for Automatic Short Answer Grading Evaluation," in *Proceedings of the Twelfth Language Resources and Evaluation Conference*, Marseille, France, Feb. 2020, pp. 2634–2643.

[38] L. B. Galhardi, "PT_ASAG_2018." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/lucasbgalhardi/pt-asag-2018.

[39] W. Antoun, F. Baly, and H. Hajj, "AraBERT: Transformer-based Model for Arabic Language Understanding." arXiv, Mar. 07, 2021, https://doi.org/10.48550/arXiv.2003.00104.

[40] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic Attribution for Deep Networks," in *Proceedings of the 34th International Conference on Machine Learning*, Jul. 2017, pp. 3319–3328.

[41] Y. Saoudi and M. M. Gammoudi, "A Comprehensive Review of Arabic Question Answering Datasets," in *Neural Information Processing*, Singapore, 2024, pp. 278–289, https://doi.org/10.1007/978-981-99-8126-7_22.

[42] L. Ouahrani and D. Bennouar, "Paraphrase Generation and Supervised Learning for Improved Automatic Short Answer Grading," *International Journal of Artificial Intelligence in Education*, vol. 34, no. 4, pp. 1627–1670, Dec. 2024, https://doi.org/10.1007/s40593-023-00391-w.