AFAR: Automated Feedback for Arabic Responses of Short-Answer Questions

Sara Alqaidi [1], Omaima Almatrafi [1], Arwa Wali [1]

[1] *Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia*

Abstract **– Providing feedback is a vital component of the effective learning process. It guides students in recognizing their weaknesses and addressing areas for improvement, rather than merely assigning final scores. However, providing manual feedback is often time-consuming and labour-intensive for instructors, especially when managing a large number of students. Previous studies have explored various approaches for automatic feedback generation, including rule-based systems and machine learning techniques. More recently, research has investigated the use of large language models (LLMs), such as GPT, to automate feedback generation. However, most of these works have focused on English-language responses and technical subjects. There is still limited research applying LLMs to provide feedback for short-answer tasks in low-resource languages, such as Arabic. To bridge this gap, this study introduces AFAR (Automated Feedback for Arabic Responses), a novel framework designed to generate personalized feedback on students' short-answer responses in Arabic using GPT-4. AFAR comprises prompt engineering and few-shot learning techniques to guide GPT-4 in producing meaningful feedback. To ensure quality, the feedback was assessed by two human evaluators in terms of coherence, relevance, clarity, and personalization. The results showed that the feedback generated by the GPT-4 was generally high in quality, with 90% rated as coherent, 82% as relevant, 92% as clear, and 85% as personalized.**

DOI: 10.18421/TEM153-63
https://doi.org/10.18421/TEM153-63

Corresponding author: Sara Alqaidi,
Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
Email: salqaidi0003@stu.kau.edu.sa

Received: 26 August 2025.
Revised: 09 February 2026.
Accepted: 04 March 2026.
Published: 27 August 2026.

Keywords **– Prompt-based learning, Generative artificial intelligence, feedback quality, evaluation.**

1. Introduction

In education, providing effective feedback is a crucial component of the teaching and learning process. High-quality feedback supports students in identifying misconceptions, bridging knowledge gaps, and improving overall academic performance. However, instructors often face challenges in providing timely and constructive feedback, especially when managing large cohorts of students and limited teaching resources. Furthermore, the process of providing feedback becomes increasingly resource-intensive for constructed response questions, such as short answer questions, which require individual analysis and interpretation of each student's response [1]. To alleviate this burden, automated feedback generation systems have emerged as a potential solution.

Automatic feedback systems leverage advanced technologies such as Natural Language Processing (NLP), Machine Learning, and Generative AI. These systems can handle different types of constructed answers, from simple short-answer questions to more in-depth explanations, providing timely, relevant, and personalized feedback. Previous studies explored various automated feedback systems for educational tasks, such as generating code explanations for beginner programmers [2], [3] and responding to forum posts to support learners in Massive Open Online Courses [4]. For example, the study by [3] developed an adaptive immediate feedback system to provide high school programmers with real-time feedback. Despite being promising, traditional feedback systems, such as machine learning and rule-based systems, can be expensive to develop, as they often require large volumes of historical, domain-specific data for training [1], [5]. To overcome this issue, recent research has explored the use of large language models (LLMs) to provide customized feedback without requiring extensive input.

LLMs have learned language patterns from large amounts of data and have been able to produce human like text. Their advanced language processing capabilities make them valuable across multiple domains, including education, healthcare, and software development. In educational settings, several studies explored the use of LLMs, particularly to generate feedback on project proposals [6], short answer questions [7], [8], [9], essays [6], [9], and programming assignments [10], [6], [11], [1]. Previously, LLMs were used for grading or providing feedback on short answers and essays with pre-trained language models such as Bidirectional Encoder Representations from Transformers (BERT) [12], [13], [14]. However, these models required extensive fine-tuning to achieve satisfactory performance. With advancements in LLM capabilities, recent studies have shifted toward using more powerful models, such as the Generative Pretrained Transformer (e.g., GPT-4), for generating feedback [15], [7].

GPT is a type of large language model (LLM) developed by OpenAI, designed to generate human-like text based on input received [6]. It can achieve high performance when designing well-crafted prompts to generate the desired output. Despite GPTs being widely and increasingly used for educational tasks, few studies have been conducted on their effectiveness with the Arabic language, particularly in providing short answer feedback. Previous studies, such as those of [16], have shown that large multilingual models perform worse on several benchmarks than task-specific Arabic models, especially in zero- and few-shot scenarios. This highlights the importance of testing general-purpose LLMs in real-life Arabic learning scenarios. This work aims to narrow this gap by examining the ability of GPT-4 to provide insightful, context-aware, and educationally sound feedback on Arabic short-answer questions.

A key distinction of this work, compared to prior research, is its emphasis on short answer questions. Short-response questions differ from other types of assessment tasks due to their distinct characteristics and the cognitive demands they place on both students and grading systems. Unlike fixed-response formats such as fill-in-the-blank questions, which typically require recognition or recall of specific terms, short-answer questions often require students to generate responses using external knowledge and apply critical thinking skills [17]. Responses to short-answer questions are typically written in natural language, ranging from a single phrase to a short paragraph, with the focus placed on content accuracy rather than writing style [18]. This makes short-answer questions more targeted than essays, which assess broader argumentation and writing skills across multiple paragraphs, while still offering a more open-ended evaluation of knowledge than rigid formats like multiple-choice or closed questions.

Despite the educational importance of short-answer questions, a notable gap remains in research on automated feedback generation in Arabic. Arabic has several unique characteristics that distinguish it from other languages, such as English, which have been more extensively studied and investigated. Particularly, when it comes to natural language processing (NLP) and feedback generation. The language has a lot of different forms, including complex affixation, templatic structures, and root-based word formation. These linguistic features make it more difficult for AI systems to accurately understand and generate Arabic text, reducing the effectiveness of general-purpose large language models (LLMs) [19]. Such models are typically trained on English or multilingual corpora that fail to capture the structural diversity and semantic complexity of Arabic adequately. In addition, Arabic remains a low-resource language in the field of NLP, and it does not have easy access to a large number of annotated datasets. While English benefits from a wide range of benchmarks and datasets in this area, Arabic lacks comparable resources, making the training, evaluation, and deployment of Arabic-capable LLMs especially challenging. Moreover, there are not many models that have been tested in Arabic feedback tasks, and most of them only do classification instead of generative feedback. Therefore, examining and testing GPT-4's ability to provide formative feedback on Arabic short-answer questions fills a significant gap in the research. This paper introduces the **AFAR** (**A**utomated **F**eedback for **A**rabic **R**esponse) framework, designed to harness GPT as a reasoning tool. The aim is to explore the capabilities of GPT-4 for generating feedback on students' written responses to short-response questions in Arabic using prompt engineering and few-shot learning techniques. Initially, various prompt phrasings with different instructional formats were designed to evaluate the effectiveness of GPT in generating feedback. The most effective prompt-based feedback was selected and integrated into a few-shot learning setup to guide the model toward producing feedback aligned with the desired structure and tone.

The main contributions of this paper are: 1) an AFAR framework is proposed to exploit GPT-4's generative capabilities to produce effective feedback on Arabic short-answer responses, 2) Two strategies are employed to enrich GPT in generating high-quality feedback, 3) Experimental results and human evaluations confirm that the proposed method can deliver high-quality feedback without the requirement for extensive annotated data.

This study explores the following research question:

To what extent can GPT-4 provide high-quality feedback on the students' short-answer responses in Arabic?

The rest of this paper is organized as follows: Section 2 summarizes the related works. Section 3 provides details about the methodology used, including data sources, and explicates the details of the proposed framework. Section 5 shows the experimental results along with a thorough analysis. Lastly, Section 6 concludes the paper and discusses limitations and future work.

2. Literature Review

Providing feedback has become an increasingly important aspect of learning and teaching systems, emerging as a vital area of research that has gained significant attention in recent years. Several approaches have been proposed for automatic feedback generation, ranging from rule-based to more sophisticated Machine Learning (ML) and LLM techniques. This section covers these three approaches and their limitations.

2.1. Rule-Based and Machine Learning Techniques

Researchers initially utilized rule-based or classical machine learning models to look into automatic feedback and grading. These systems could be understood, but they often could not be scaled up or changed to handle complex student input. An early study by [20] utilized a rule-based approach to provide feedback. It allowed instructors to provide feedback on student learning behaviors by establishing conditional rules based on student activities, such as lesson attendance and academic performance. To address some limitations of rule-based systems, clustering and ML-based feedback systems emerged, offering greater flexibility in feedback reuse and efficiency. The study by [21] aimed to develop a model for automatically grading short answer questions and providing helpful feedback to students' answers, especially for the UK's GCSE system. In order to categorize student responses based on similarities, they utilized a clustering technique, which improved grading speed and consistency by enabling common feedback across each cluster. They used the K-means algorithm to cluster student answers; each cluster received feedback that was tailored to them, giving students insights into typical mistakes and strong points in their responses. In addition, a study by [22] proposed a model for an automatic grading system and provided personalized feedback to students for essay-type answers. They utilized NLP and machine learning techniques. Their methodology hinged on neural networks, specifically using LSTM and BiLSTM networks to extract and analyze the semantic meanings of answers.

Based on that, the proposed system partitions the answers into clusters using the k-means algorithm to facilitate providing feedback. Rule-based and traditional Machine Learning offer structured and understandable answers, but at the same time they depend on custom features and rules that are specific to the domain. This makes them difficult to scale or adapt to new tasks, domains, or languages such as Arabic. This limitation has made it possible for models with more flexibility, such as large language models, to overcome these challenges. A more recent study by [23] aimed to develop and evaluate a system for automatic feedback generation for short-answer questions. They introduced the Answer Diagnostic Graph (ADG), a graph-based representation of the logical structure of the prompt text and the model answer. The algorithm creates personalized feedback by matching justification cues from a student's response to ADG nodes using Sentence-BERT and selecting a feedback template based on their relationship to the model response. These templates were constructed by analyzing common error types in the actual responses of the students and creating generic feedback patterns. The effectiveness of the system was tested with 39 Japanese high school students, where participants alternated between receiving official answer explanations and system-generated feedback. The results of the questionnaire showed that the proposed feedback system improved the understanding of the students' mistakes, grasp of key points, and motivation to revise, although some participants considered it vague.

2.2. Potential of Large Language Models

Following great advances in NLP techniques, researchers began integrating large language models (LLMs) such as GPT to generate more personalized and human-like feedback. Research conducted by [6] investigated whether ChatGPT might be used to give students feedback on open-ended writing projects like project proposals and essay assignments. The study analyzed feedback on 103 project proposals from a postgraduate data science course. They evaluated ChatGPT feedback for readability, alignment with instructor feedback, and effectiveness. In order to measure alignment with instructor feedback, metrics like precision and recall were used. Interestingly, they found that ChatGPT-generated feedback was significantly more readable than instructor feedback. There was some work on programming tasks that showed promise for LLMs in both grading and providing feedback for structured tasks such as code snippets. A paper authored by [11] explored the capabilities of GPT-4 Turbo in generating formative feedback for programming exercises.

They selected two assignments (SimpleWhileLoop and Queue) from an introductory programming course and randomly chose 55 student programming submissions for this study. The prompts contain both the programming task specification and a student's submission as input. They evaluated the quality of generated feedback using qualitative thematic analysis based on correctness, personalization, and fault localization. The result indicates that GPT-4 Turbo demonstrated significant improvements, such as more structured and consistent feedback, compared to GPT-3.5 in previous work. In addition, [1] examined the use of GPT-4 for generating formative feedback on student programming submissions in an algorithms and data structures course. It evaluates 113 student submissions across four programming problems, leveraging few-shot learning to guide GPT-4 in assessing conceptual understanding and syntax. As a result, GPT-4 achieved a high accuracy of over 90% and provided constructive feedback tailored to students' needs. Furthermore, a study by [7] proposed a framework that uses the capabilities of ChatGPT to generate explainable feedback and assessments for student responses. This approach involves prompting ChatGPT with three strategies to extract rationales: simple instructions, complex instructions, and example instructions. These rationales are used to fine-tune a smaller language model (Long T5), creating an interpretable automated assessment system.

The results indicate that ChatGPT-generated rationales using example instruction, which leverages few-shot learning, proved most effective. Moreover, the study by [24] evaluated the capacity of LLM-based tools, specifically GPT-3.5 and PaLM 2, in providing feedback on student assignments in a concurrent programming course. They analyzed 52 student Java submissions and conducted four experiments using different prompt configurations to assess the models' ability to detect the errors without executing the code. The findings show that although the tools often detected errors accurately, their overall accuracy rate was 50%, with poor precision and a propensity to over-flag problems, which could result in unclear feedback. While LLMs have seen widespread adoption across diverse domains, their application in generating constructive feedback, especially for short-answer questions, has only recently begun to gain research attention.

3. Methodology

In this section, the methodology underlying the proposed framework AFAR (Automated Feedback for Arabic Responses) is presented, as illustrated in Figure 1. The framework consists of three main phases: dataset preparation, generating a few-shot example, including prompt engineering, and automated feedback generation.

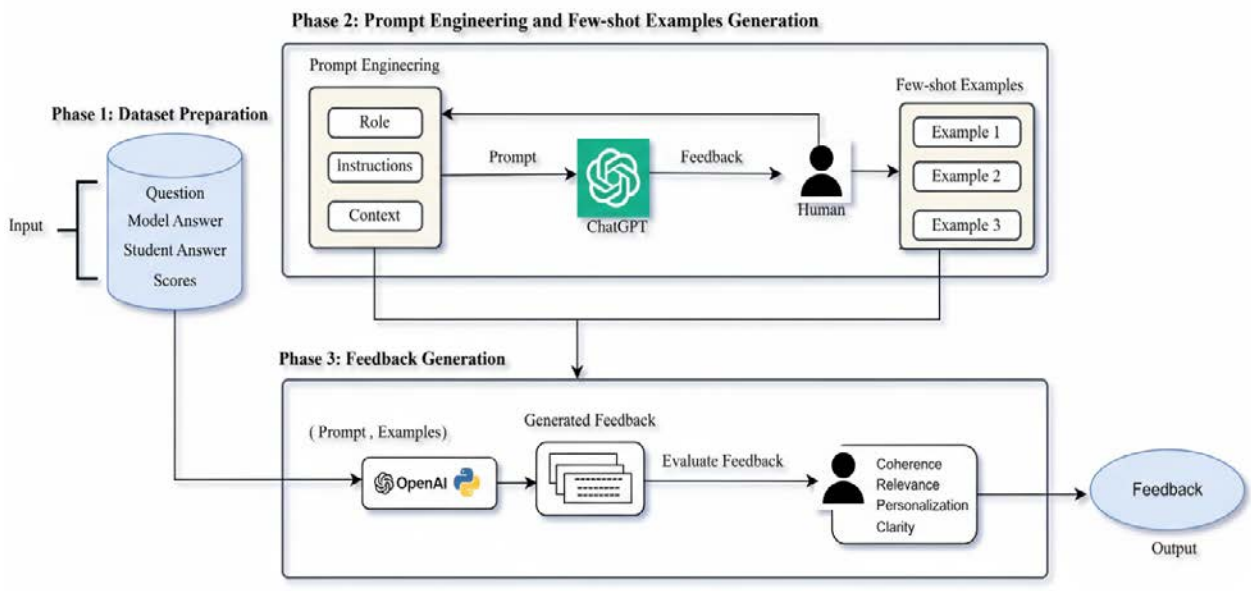


Figure 1. Architecture of AFAR: The proposed framework for automated feedback generation on Arabic short answers

3.1. Phase 1: Data Collection and Preparation

To fulfill research objectives, the Arabic Short Answer Grading (AR-ASAG) dataset was selected for this purpose [25]. It consists of approximately 2133 samples spanning 48 distinct questions from a cybercrime course. Each question had corresponding student responses, model (reference) answer, and average scores on a rating system of 1 to 5 determined by two human annotators.

The average grade of the two experts makes AR-ASAG the gold standard for comparing the results of the proposed model. Table 1 shows the sample of the AR-ASAG dataset. In this study, four student responses per question were randomly selected, resulting in a representative subset of 193 samples for analysis.

Table 1. Sample of AR-ASAG dataset

Question	Define the term of cybercrime? عرف مصطلح الجريمة الالكترونية؟
Model Answer (MA)	Cybercrime refers to any illegal activity carried out using electronic devices such as smartphones, computers, or the internet. It typically involves the offender gaining financial or personal benefits while causing harm or loss to the victim. Most cybercrimes aim to hack, steal, or destroy information, with the internet often serving as the main tool or environment for such acts. هي كل سلوك غير قانوني يتم باستخدام الأجهزة الإلكترونية مثل الهاتف، او الكمبيوتر, او الانترنت و ينتج عنه حصول المجرم على فوائد مادية وغالبا ما يكون هدف هذه الجرائم هو القرصنة من أجل سرقة أو أو معنوية مع تحميل الضحية خسارة إتلاف المعلومات, وتكون عادة الانترنت أداة أو مسرحا لها.

Students Answers (SA)	**Expert 1 mark**	**Expert 2 mark**	**AVG**
Sample 1: It is illegal behavior using electronic means that results in the criminal achieving material or moral objectives. The goal of these crimes is often hacking. هي سلوك غير قانوني تستخدم الوسائل الالكترونية ينتج عنها حصول المجرم على أهداف مادية أو معنوية، غالبا يكون هدف هذه الجرائم هي القرصنة أي سرقة واتلاف المعلومات	3	5	4
Sample 2: It is an unethical behavior carried out electronically, aiming to gain financial gain and causing harm to the victim. هي سلوك غير أخلاقي يتم عن طريق وسائل الكترونية يهدف الى عائدات مادية و يسبب اضرارا للضحية	2.5	3.5	3

Figure 2 shows the grade distribution for the selected sample.

For the preparation phase, missing or null values were reviewed in the dataset to guarantee accurate and legitimate results.

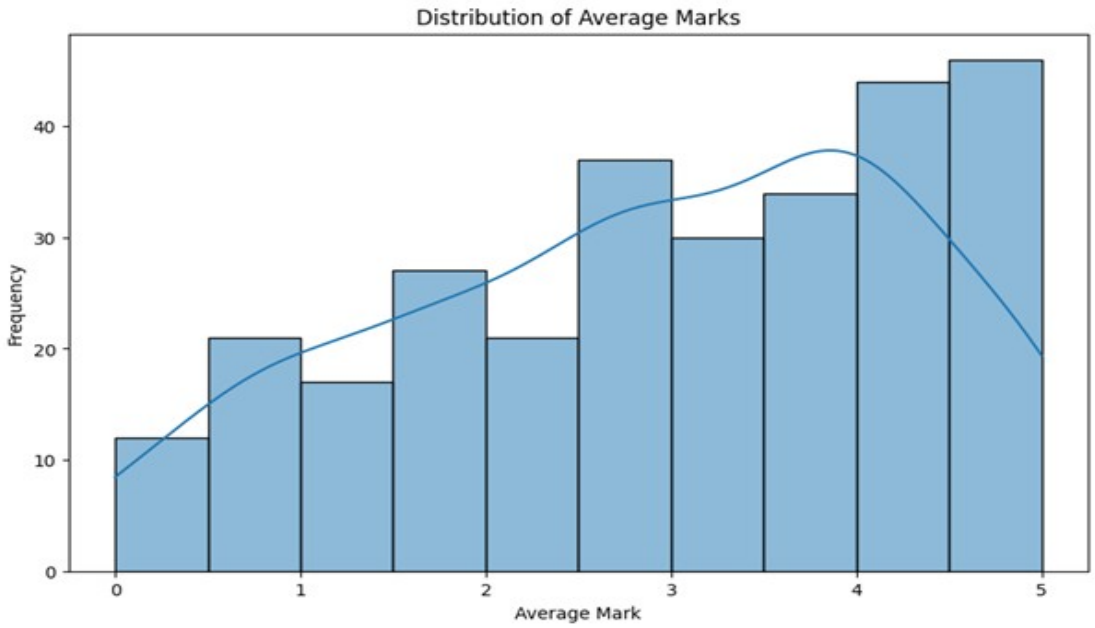


Figure 2. Grade distribution

3.2. Phase 2: Generating a Few-Shot Example

To effectively leverage LLMs such as GPT, it is essential to adopt a structured approach that adapts the model's general capabilities to the specific requirements of formative assessment. Given the model's sensitivity to input phrasing and its reliance on contextual cues, this phase includes two key components: **prompt engineering** and **few-shot learning**. The following sections elaborate on each of these components in detail.

3.2.1. Prompt Engineering

Prompt engineering refers to the process of crafting and refining input prompts to enhance the output of large language models (LLMs) such as GPT [26]. So, the key benefit of prompt engineering is the ability to obtain optimized outputs with minimal post-generation effort. Thus, the quality of the output depends mainly on formulating appropriate prompts that direct the model to generate the desired output. One of the essential tactics in prompt engineering is the role of prompting, which instructs the model to act as if it holds a specific function or expertise. For example, by positioning the model as a "teacher" or "feedback assistant", the model adapts its output style, tone, and vocabulary to match the assumed identity. This approach uses models like GPT-4, which are trained to follow instructions and respond according to specific roles. In this study, this technique was used to ensure that the feedback generated by the model mimicked the style of constructive, human-like commentary given by educators, particularly in formative settings.

Technically speaking, prompts in large language models such as GPT-4 would be passed as sequences of messages with roles such as system, user, and assistant. This is part of the chat-based API architecture used in the GPT-4 model via OpenAI's chat completions endpoint. While the user message provides the task content, the system message typically defines the model's identity and task rules. The system prompts were designed to define the educational context, advise the model to emphasize kindness and clarity, and stress that feedback needs to be given in Arabic. The total token length (prompt, input, and expected output) was tracked to prevent going over the model's limit because prompt behavior is sensitive to token count. For GPT-4, the token limit was 128,000 tokens, each token corresponding roughly to four characters. Exceeding this limit can result in incomplete outputs or degraded model performance, particularly when key prompt instructions are placed too far back in the input sequence. This called for careful engineering of prompt phrasing to be brief but detailed enough to direct the output structure and content. During the implementation of AFAR, prompt engineering was not a one-time configuration but was actually an iterative process aimed at optimization. The prompt wording was refined across multiple cycles of testing and analysis. Each iteration involved adjusting sentence structure, emotional cues, verbosity, and placement of instructions within the system prompt. For example, placing tone-related instructions such as ("use a supportive and encouraging tone") at the beginning of the prompt resulted in more emotionally aligned output compared to placing them near the end.

Multiple alternative prompt phrasings in Arabic were also examined, testing imperative forms (e.g.,"ساعد الطالب", help the student) versus descriptive framing (e.g.,"مهمتك تقديم تغذية راجعة", your task is to provide feedback) to improve adherence to task requirements. These empirical refinements show that prompt engineering is both a technical and language-based task that significantly influences the reliability of LLM outputs.

During the iterative process to optimize GPT-4's performance in generating effective feedback for Arabic short-answer questions, the model was initially prompted to assess the correctness of student responses against a reference answer and to identify key weaknesses. While this initial strategy yielded factually accurate outputs, it often lacked the depth and instructional value needed for meaningful feedback. It was observed that too general or ambiguous prompts would often produce incomplete or overly general feedback. To address this, the prompts were refined to include specific instructional roles such as feedback generator, emotional tone, and clear expectations such as length and structure. This refinement process enhanced clarity, promoted a supportive tone, and ensured better alignment with educational objectives, resulting in more consistent and personalized outputs.

Recognizing the variability in student response quality, completeness, and correctness, the model's ability to generate tiered feedback was evaluated. This approach involved tailoring the feedback based on the completeness and correctness of each answer. Specifically, the model was prompted to 1) Provide improvements and supportive suggestions for entirely incorrect answers, and 2) Identify and explain weak aspects in partially correct responses while offering guidance to address misconceptions. This is called "tiered feedback". It helps students to get exactly the support they need, depending on how much they already understand [27].

As part of this refinement, the prompt was also designed to instruct the model to include rationales within its feedback, ensuring that responses maintained a positive and supportive tone. GPT-4 was prompted to clarify the reasons for its comments. This is called giving a "rationale". By providing these rationales, students can understand the reasoning behind their evaluation rather than simply being informed of their scores. Students are more likely to learn from mistakes if they know why they were wrong or how to improve their answer. Studies have shown that giving students feedback with explanations helps them fix how they think and do better next time [28]. When people answer open-ended questions, feedback that does not explain why it was given can feel vague and useless.

There was also an effort to make the feedback sound positive and encouraging. Too much negative feedback can make students give up or feel stressed, especially if they do not know much about the subject to begin with. It has been shown that a supportive tone would make the students more motivated, surer of themselves, and more likely to use feedback to get better [25]. Accordingly, the model was constrained to avoid harsh language and instead emphasize constructive and positive aspects. To guarantee linguistic alignment with the dataset, the prompt was written in Arabic. Thus, the prompt was designed as follows: "You are an automatic generator of short answer feedback. If you were given the student's answer, reference answer, and grade, could you please analyze the weaknesses in the student's response and offer constructive and positive feedback in 3-4 lines? Ensure that the feedback is directed to the student without evoking negative emotions". The prompts were carefully developed with particular objectives in mind to help GPT-4 produce high-quality feedback. Each part of the prompt was selected carefully to make the feedback more useful and clearer. These decisions were made based on ideas backed up by research in the field of education. The prompts shaped GPT-4's responses toward more educationally aligned outputs [26]. This phase was essential in setting the foundation for incorporating examples in the few-shot learning approach.

3.2.2. Few-Shot Learning

Few-shot prompting is an effective approach that allows pre-trained models to perform specific tasks and identify underlying patterns by presenting only a few examples [27]. This can be accomplished by giving the prompts a few examples to determine how LLM should respond to similar tasks [28]. This is particularly advantageous when the task requires minimal guidance or a specific format, avoiding the need for numerous examples. It can be applied to various tasks, including text generation, answering questions, and summarization. In the context of education, [29] used few-shot prompts to train GPT to solve math problems. They found that GPT achieved high accuracy in solving the issues and improved performance as the number of examples increased. In-context learning, a mechanism by which the model generalizes task behavior from examples given within the prompt itself [30], is how few-shot learning works in large language models such as GPT-3.5 and GPT4. These examples are pattern-matched at inference time rather than "learned" in the conventional sense (i.e., weights are not updated).

The model identifies the structure, tone, and logic required to reproduce comparable outputs for a new, unseen input by observing how input and output are paired across a few instances. Because of this, few-shot prompting is particularly helpful for task adaptation in situations where task-specific labeled data is limited or fine-tuning is not practical. Adding multiple input-output examples (formatted as previous turns or message blocks) straight into the model's input context is the technical way to implement a few-shot prompting. This is accomplished by generating numerous alternating user and assistant messages in OpenAI's Chat Completions API. For instance, four examples were added to the AFAR framework, where each assistant message contained the anticipated feedback and each user message contained the question, reference answer, student answer, and grade information. To construct these examples, a custom-designed prompt (as outlined in the previous section) was used along with the question, the student's answer, the reference answer, and the assigned grade. Then, ChatGPT-4 was instructed to generate feedback for 20 student responses based on this input. Based on the analysis of the generated feedback, four feedback examples were selected as reference outputs. These examples were chosen from different types of questions and different grade levels. These carefully selected examples were incorporated to guide the model in generating consistent and appropriate instructional feedback aligned with varying performance levels, as they closely matched the desired structure, tone, and relevance to the reference answers. This structure enables the user to implicitly infer the task pattern from the examples and apply it to the subsequent query, which appears as a final task message requesting feedback on a new student's answer.

3.3. Phase 3: Feedback Generation

During this phase, the GPT-4 model was utilizedvia OpenAI's Chat Completions API in completion mode to generate feedback for Arabic short-answer responses. GPT-4 was specifically selected due to its advanced capabilities in instruction following, interpretation, and structured language generation. Completion mode was preferred over chat mode, because it interprets the input as an uncompleted text or document and extends it by predicting the most probable continuation. Compared to conversation mode, this method is more effective since it enables the creation of structured and context-aware feedback. The implementation was carried out using the OpenAI Python SDK (openai package), which simplified prompt construction and API integration.

Each API call was made using the openai.ChatCompletion.create() method, specifying the model name ("gpt-4"), and providing the prompt as a sequence of role-defined messages (system and user). The input provided to the model included a system instruction that defined the model's role as a supportive tutor, along with a user message that contained the question, the reference answer, the student's response, and the assigned grade. This input format enabled the model to assess student performance and generate concise, pedagogically relevant feedback. The output was designed to be a short paragraph in 3–4 lines that acknowledged the student's effort, identified key errors or misconceptions, and offered constructive suggestions for improvement in a clear and encouraging tone. The model was configured with the following parameters: **temperature set to 0.0** for deterministic output, and **max tokens set to 500** to control the response length. The **top-p parameter was set to 1.0**, although it had minimal effect due to the zero-temperature setting. These settings facilitated effective use of GPT-4 for generating consistent and high-quality feedback, without relying on additional model training.

3.4. Human Evaluation

This section presents further details of the human evaluation conducted in this study.

- **Data Selection:** A total of 193 samples were randomly selected from the dataset, with four student answers chosen per question.
- **Annotators:** To assess the generated feedback, two annotators have been selected for the evaluation process. Both evaluators are undergraduate students at King Abdulaziz University who are Arabic speakers and have passed an information security course. Each annotator spent about five hours completing the evaluation process.
- **Evaluation Metrics:** To assess the quality of the generated feedback, the evaluation metrics proposed by [31] is used. Each was defined as follows:
  - **Coherence:** The organization and logical flow of the feedback.
  - **Relevance:** The extent to which the feedback captured and addressed the main points of the student's response compared to the correct answer.
  - **Clarity:** The ease with which the feedback can be understood.
  - **Personalization:** The degree to which the feedback is specifically tailored to the individual student's response, considering their strengths, areas of improvement, and specific context.

- **Evaluation tool:** As illustrated in Table 2, the evaluation process was constructed and organized using Google Sheets as the primary tool. The spreadsheet was designed with four distinct columns, each representing one of the evaluation metrics: Coherence, Relevance, Clarity, and Personalization. Each annotator assessed the quality of the generated feedback based on these metrics, assigning scores on a Likert scale ranging from 1 to 5, where higher values indicated stronger alignment with the desired qualities. This setup facilitated a systematic comparison among raters and provided a structured framework for assessing inter-rater agreement.

Table 2. Human evaluation scores for feedback

Coherence	Relevance	Clarity	Personalization
5	5	5	5
5	4	5	5
5	5	5	5
4	5	4	5
4	4	5	4
5	5	5	5
5	3	4	5
5	3	5	4
3	3	5	3
3	2	4	4
2	4	4	5
5	5	5	5
3	1	3	2
3	1	3	1

- **Agreement:** In this study, the agreement rate was used as a simple metric to quantify how often two annotators agree in their assessments. It provides a direct measure of the consistency between the two raters, without accounting for chance agreement. Although raw agreement does not account for chance agreement, it remains a useful preliminary indicator of annotator consistency [34]. For analysis, a simplified version of raw agreement was used by counting both exact matches and near matches, where the rating difference was within one point to reflect an acceptable tolerance in subjective judgment. Considering the ordinal nature of the ratings, for example, on a scale of 0 to 5, the concept of exact agreement was expanded to encompass near agreement. This approach is in line with previous research in NLP evaluation, where minor discrepancies in human ratings are considered acceptable due to the subjective nature of the task [32], [33].

4. Results

This section reports the experimental results, focusing on two main aspects: the level of agreement among human evaluators and the evaluation of proposed feedback generation approach within the AFAR framework.

4.1. Human Agreement Results

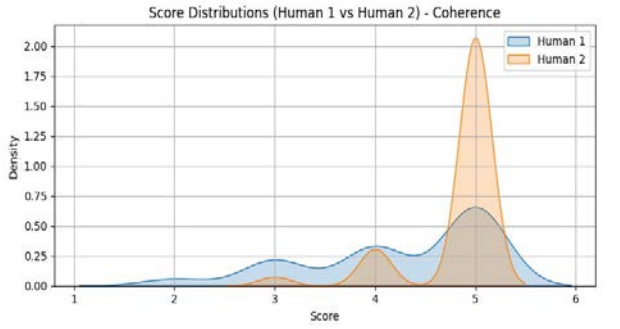
For evaluation, the simple raw agreement was computed by comparing the scores assigned by each annotator across all metrics for the same set of sample responses. As shown in Table 3, coherence achieved a moderate agreement rate of 79.7%, indicating that while the raters are reasonably aligned, there is still room for improvement due to minor inconsistencies in judgment.

Table 3. Inter-rater agreement results

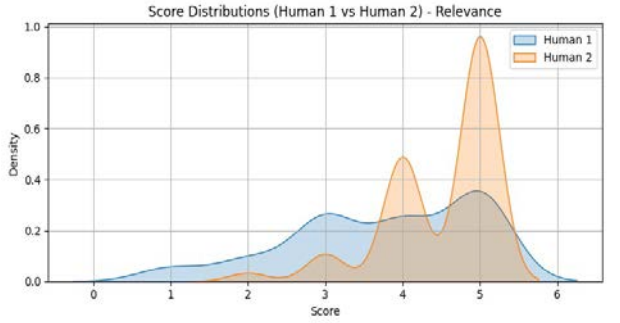
Metric	**Agreement Rate**
Coherence	79.7%
Relevance	69.8%
Clarity	88.5%
Personalization	87.0%

Relevance had the lowest agreement rate (69.8%), which means that there was a difference of more than one point between the ratings. In contrast, clarity and personalization demonstrated high agreement rates of 88.5% and 87%, respectively, indicating strong consistency between the evaluators in recognizing clear and personalized feedback.

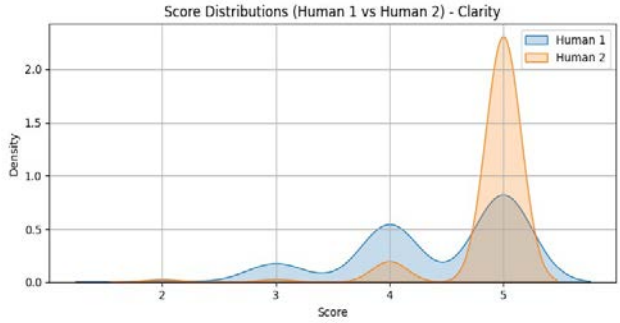
Figure 3 presents the score distribution across the four-evaluation metrics, based on the ratings provided by the two human evaluators. The distribution of scores across the evaluated criteria reveals notable discrepancies between the two human evaluators. In general, the first evaluator tends to adopt a stricter and analytical approach, utilizing a broader scoring range from 1 to 5 across all metrics. This indicates that he used a more precise methodology in the evaluation, in which he considered the differences in clarity and logical sequence of the feedback results. In contrast, the second evaluator demonstrated a tendency toward leniency, with scores often clustered closely around 5. These differences were most observed in relevance, coherence, and clarity metrics, where the first evaluator demonstrated greater sensitivity to structural and semantic nuances. However, in the personalization criterion, the two evaluators demonstrated stronger agreement, with both assigning scores mostly in the range of 4 and 5. This difference may be due to the varying sensitivity of the two evaluators to coherence flaws, which indicates the need for clearer standards or evaluation metrics to judge the structural quality of the feedback.



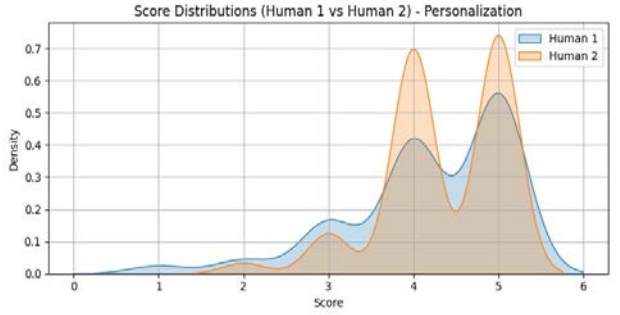
(a) Coherence



(b) Relevance



(c) Clarity



(d) Personalization

Figure 3. Score distribution of the two evaluators' results; (a) Coherence; (b) Relevance; (c) Clarity; (d) Personalization

4.2. AFAR Evaluation Result

This section presents the results of applying the AFAR framework to generate automated feedback on Arabic short-answer responses using GPT-4. Given the high level of agreement among human evaluators, the mean and standard deviation (SD) of the evaluation scores are reported to assess the overall quality of the proposed framework.

To evaluate the quality of the generated feedback, a sample of 193 student responses was assessed by two human evaluators using these four key metrics: coherence, relevance, clarity, and personalization. For each metric, the mean and standard deviation (SD) of the scores assigned by the evaluators were computed, as shown in Table 4.

Table 4. Evaluation results

Metric	Mean	SD
Coherence	4.5(90%)	0.51
Relevance	4.1(82%)	0.74
Clarity	4.6(92%)	0.64
Personalization	4.2(85%)	0.45

To support a meaningful interpretation, the average scores were categorized into qualitative ranges: 4.5–5.0 as Excellent, 4.0–4.49 as Good, 3.0–3.99 as Fair, 2.0–2.99 as Poor, and 1.0–1.99 as Very Poor. The result revealed that, on average, 90% of the feedback was rated as coherent, indicating that most responses maintained a clear and organized structure. On average, 82% of the feedback was considered relevant, demonstrating that the model generally aligns its responses with the given input. However, it was the lowest score among the four evaluated metrics. For the clarity metric, the average score was 92%, indicating that most feedback responses were articulated in a clear and concise manner. The average personalization score was 85%, suggesting that while the model often adapts the feedback to the specific answer, there are instances where the responses are more generic or lack context sensitivity.

5. Discussion

A descriptive analysis was performed to assess the quality of the feedback generated in four dimensions rated by humans: clarity, coherence, personalization, and relevance. The findings revealed that clarity had the highest mean score, indicating understandable feedback. Coherence also received high ratings, reflecting a logical flow of content. Personalization was slightly lower, but still strong, showing that the model often tailored feedback to individual responses. However, relevance had the lowest average score, suggesting that feedback sometimes failed to fully address the key points or intent of the student's answer. For instance, a student's response to the question "Define the term cybercrime?" was: "It is illegal behavior using electronic means that results in the criminal achieving material or moral objectives. The goal of these crimes is often hacking". This response has a model answer of "Cybercrime refers to any illegal activity carried out using electronic devices such as smartphones, computers, or the internet.

It typically involves the offender gaining financial or personal benefits while causing harm or loss to the victim. Most cybercrimes aim to hack, steal, or destroy information, with the internet often serving as the main tool or environment for such acts." The feedback generated was as follows: "Well done! You have demonstrated a solid understanding of the concept of cybercrime by referring to illegal behavior and material and moral objectives. To further enrich your answer, you could focus on explaining that cybercrime includes the use of electronic devices in general, not just hacking, and that the internet is often the primary vehicle for these crimes. This will help improve the accuracy and depth of your answer." This feedback example was rated as 3 out of 5 in the relevance metric, the degree to which the feedback addressed the key points of the student's response compared to the correct answer. Feedback does not address all possible misunderstandings in the student's answer. For example, the student did not mention that hacking involves theft and destruction of information, which is part of the correct answer. Although the feedback generated by the GPT-4 model was generally strong and performed well across most assessment metrics, there was a slight decline in relevance (82%). This suggests that the model did not always address all the weaknesses in students' answers. Sometimes, the feedback was too general or did not focus precisely on the core problem in the student's answer. These results point to the importance of a more precise alignment between what the model is asked to produce and the reference answers, perhaps through the use of task-specific feedback templates, more specific knowledge-based instructions, or even the integration of guided retrieval components within the prompt to improve the model's understanding of context and content.

6. Conclusion

This paper proposed a framework called AFAR, which leveraged the few-shot learning and prompt engineering techniques to support GPT-4 for generating automated feedback on Arabic short answer responses. The experimental findings suggest that, although GPT is capable of producing free-form explanations through natural language instructions, an AFAR, based on example-guided prompting, achieves the best performance. The extensive experiments and human evaluation results confirm that the proposed strategy can deliver high-quality feedback in terms of coherence, relevance, clarity, and personalization.

However, this study has several limitations that should be acknowledged. First, the design of the prompt may vary across individuals, leading to inconsistencies influenced by subjective decisions regarding phrasing.

Second, although the annotators received appropriate guidance, their limited background in exam assessment may have influenced the quality of the evaluations. Third, the use of a fixed prompt phrasing and the limited number of "few-shot" examples may limit the model's ability to adapt to different student response styles. Future studies are encouraged to explore more adaptable and flexible strategies for designing prompts or using "prompt tuning" techniques to improve the model's ability to generalize and adapt to diverse learning contexts. One effective idea that could be implemented in the future is to utilize an interactive approach to modifying prompts, rather than relying on fixed templates. Instead of the model operating the same way for every answer, the prompt could be customized based on the type of student's answer or the learning objective of the question. This could make feedback more personalized, more useful, and better aligned with the assessment objective. Additionally, further research is needed to expand the feedback evaluation process to include various annotators, such as professional teachers or subject matter experts. Having assessors from diverse fields would help to ensure that the assessment is fairer and realistic. Additionally, it will ensure that the quality of feedback is measured not only from a technical perspective but also from an educational perspective. Finally, to better assess the efficacy of feedback based on LLMs, future research should compare learning outcomes between instructor-written feedback and output generated by the GPT. An AFAR framework contributes significantly to the research community since it provides a replicable and practical approach for leveraging prompt engineering and few-shot learning in low-resource languages such as Arabic, filling a significant gap in the NLP field. In addition, AFAR presents a scalable and effective solution for delivering high-quality, individualized feedback to students. Its ability to generate coherent, relevant, clear, and personalized feedback can support formative assessment, reduce the burden on educators, and promote more consistent feedback practices in large-scale or online learning environments.

References:

[1]. Nguyen, H., & Allan, V. (2024). Using GPT-4 to provide tiered, formative code feedback. *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, 958-964.

[2]. Price, T. W., Dong, Y., & Lipovac, D. (2017). iSnap: towards intelligent tutoring in novice programming environments. *Proceedings of the 2017 ACM SIGCSE Technical Symposium on computer science education*, 483-488.

[3]. Marwan, S., et al. (2020). Adaptive immediate feedback can improve novice programming engagement and intention to persist in computer science. *Proceedings of the 2020 ACM conference on international computing education research*, 194-203.

[4]. Li, C., & Xing, W. (2021). Natural language generation using deep learning to support MOOC learners. *International Journal of Artificial Intelligence in Education*, *31*(2), 186-214.

[5]. Keuning, H., Jeuring, J., & Heeren, B. (2018). A systematic literature review of automated feedback generation for programming exercises. *ACM Transactions on Computing Education (TOCE)*, *19*(1), 1-43.

[6]. Dai, W., et al. (2023). Can large language models provide feedback to students? A case study on ChatGPT. *2023 IEEE international conference on advanced learning technologies (ICALT)*, 323-325.

[7]. Azaiz, I., Kiesler, N., & Strickroth, S. (2024). Feedback-generation for programming exercises with gpt-4. *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*, 31-37.

[8]. Sung, C., et al. (2019). Pre-training BERT on domain resources for short answer grading. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 6071-6075.

[9]. Li, Z., et al. (2023). Learning when to defer to humans for short answer grading. In *International Conference on Artificial Intelligence in Education*, 414-425. Springer Nature Switzerland.

[10]. Condor, A. (2020). Exploring automatic short answer grading as a tool to assist in human rating. *International Conference on Artificial Intelligence in Education*, 74-79. Springer International Publishing.

[11]. McGowan, A., Anderson, N., & Smith, C. (2024). The use of ChatGPT to generate summative feedback in programming assessments that is consistent, prompt, without bias and scalable. *Proceedings of the Cognitive Models and Artificial Intelligence Conference*, 39-43.

[12]. Abdelali, A., et al. (2024). LAraBench: Benchmarking Arabic AI with large language models. *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, 487-520.

[13]. Burrows, S., Gurevych, I., & Stein, B. (2015). The eras and trends of automatic short answer grading. *International journal of artificial intelligence in education*, *25*(1), 60-117.

[14]. Galhardi, L. B., & Brancher, J. D. (2018). Machine learning approach for automatic short answer grading: A systematic review. *Advances in Artificial Intelligence (IBERAMIA 2018)*, 380–391. Springer.

[15]. Antoun, W., Baly, F., & Hajj, H. (202). Arabert: Transformer-based model for arabic language understanding. *Proceedings of the 4th workshop on open-source arabic corpora and processing tools, with a shared task on offensive language detection*, 9-15.

[16]. Pardo, A., et al. (2018). OnTask: Delivering data-informed, personalized learning support actions. *Journal of Learning Analytics*, *5*(3), 235–249.

[17]. Süzen, N., et al. (2020). Automatic short answer grading and feedback using text mining methods. *Procedia computer science*, *169*, 726-743.

[18]. Ye, X., & Manoharan, S. (2019). Providing automated grading and personalized feedback. *Proceedings of the International Conference on Artificial Intelligence, Information Processing and Cloud Computing*, 1-5.

[19]. Furuhashi, M., et al. (2025). Automatic feedback generation for short answer questions using answer diagnostic graphs. *arXiv preprint arXiv:2501.15777*.

[20]. Estévez-Ayres, I., et al. (2025). Evaluation of LLM tools for feedback generation in a course on concurrent programming. *International journal of artificial intelligence in education*, *35*(2), 774-790.

[21]. Ouahrani, L., & Bennouar, D. (2020). AR-ASAG an Arabic dataset for automatic short answer grading evaluation. *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 2634-2643.

[22]. Liu, P., et al. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM computing surveys*, *55*(9), 1-35.

[23]. Shute, V. J. (2008). Focus on formative feedback. *Review of educational research*, *78*(1), 153-189.

[24]. Lipnevich, A. A., & Smith, J. K. (2009). "I really need feedback to learn:" students' perspectives on the effectiveness of the differential feedback messages. *Educational Assessment, Evaluation and Accountability*, *21*(4), 347-367.

[25]. Hattie, J., & Timperley, H. (2007). The power of feedback. *Review of educational research*, *77*(1), 81-112.

[26]. Jacobsen, L. J., & Weber, K. E. (2025). The promises and pitfalls of large language models as feedback providers: A study of prompt engineering and the quality of AI-driven feedback. *AI*, *6*(2), 35.

[27]. Wang, Y., Yao, Q., Kwok, J. T., & Ni, L. M. (2020). Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, *53*(3), 1-34.

[28]. Wan, T., & Chen, Z. (2024). Exploring generative AI assisted feedback writing for students' written responses to a physics conceptual question with prompt engineering and few-shot learning. *Physical Review Physics Education Research*, *20*(1), 010152.

[29]. Zong, M., & Krishnamachari, B. (2023). Solving math word problems concerning systems of equations with GPT-3. *Proceedings of the AAAI Conference on Artificial Intelligence*, *37*(13), 15972–15979.

[30]. Pezzotti, N. (2021). *GPT-3 for few-shot dialogue state tracking*. Master's thesis, University of Cambridge, Department of Engineering.

[31]. Liu, Y., et al. (2023). G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, 2511-2522.

[32]. Callison-Burch, C., Osborne, M., & Koehn, P. (2006). Re-evaluating the role of BLEU in machine translation research. *11th conference of the european chapter of the association for computational linguistics*, 249-256.

[33]. Zhang, Y., et al. (2020). Dialogpt: Large-scale generative pre-training for conversational response generation. *Proceedings of the 58th annual meeting of the association for computational linguistics: system demonstrations*, 270-278.

[34]. Artstein, R., & Poesio, M. (2008). Survey article: Inter-coder agreement for computational linguistics. *Computational linguistics*, *34*(4), 555-596.