

Evaluating LLM-Assisted Multiple-Choice Question Generation in Requirements Engineering: Prompting Strategies and Professor Validation

Camilla Borchhardt Quincozes
Faculdade de Computação (FACOM)
Universidade Federal de Uberlândia (UFU)
ORCID: [0009-0004-7398-3551](https://orcid.org/0009-0004-7398-3551)
camillaquincozes@ufu.br

Diego Nunes Molinos
Faculdade de Computação (FACOM)
Universidade Federal de Uberlândia (UFU)
ORCID: [0000-0002-7455-7624](https://orcid.org/0000-0002-7455-7624)
diego.molinos@ufu.br

Rafael Dias Araújo
Faculdade de Computação (FACOM)
Universidade Federal de Uberlândia (UFU)
ORCID: [0000-0003-0545-2519](https://orcid.org/0000-0003-0545-2519)
rafael.araujo@ufu.br

Silvio Ereno Quincozes
Faculdade de Computação (FACOM)
Universidade Federal de Uberlândia (UFU)
Universidade Federal do Pampa (UNIPAMPA)
ORCID: [0000-0001-6793-4033](https://orcid.org/0000-0001-6793-4033)
silvioquincozes@unipampa.edu.br

Gilleanes Thorwald Araujo Guedes
Universidade Federal do Pampa (UNIPAMPA)
ORCID: [0000-0001-5457-2600](https://orcid.org/0000-0001-5457-2600)
gilleanesguedes@unipampa.edu.br

Abstract

Large Language Models (LLMs) are increasingly being explored for their potential to automate the generation of multiple-choice questions (MCQs). However, the quality of these questions is still influenced by various factors, including pedagogical, conceptual, and linguistic considerations. This article presents the outcomes of a new experimental round conducted in 2026, focusing on MCQ generation within the Requirements Engineering domain, and builds upon a previous study published in 2025. The current study uses the same domain, Prompt Engineering techniques, evaluation criteria, and validation protocol, and merges automatic evaluations by LLMs with assessments from professors. In total, 200 questions were generated and analyzed. The results indicate that automatic evaluations and human assessments often produce differing judgments, with LLMs generally assigning higher scores than their human counterparts. Additionally, the findings reveal that Prompt Engineering techniques and generator models exhibit different behaviors across evaluators. Challenges related to the criteria of diversity and complexity persist in automated question generation. This study highlights the considerable potential of LLMs to enhance assessment design, while also emphasizing the essential role of human curation and the need for continuous validation in the rapidly evolving landscape of LLM ecosystems.

Keywords: Large Language Models; Multiple-Choice Questions; Prompt Engineering; Automated Question Generation; Requirements Engineering; Educational Assessment; Human Evaluation

1 Introduction

Large Language Models (LLMs), driven by advanced transformer-based architectures, are transforming various educational practices, acting not only as tools but also as true companions and virtual assistants. In content generation, they reduce educators' manual workload by automatically creating high-quality materials, such as textbooks, lesson plans, interactive resources, and even programming code and experimental models for research. Regarding material personalization, LLMs identify needs based on the student's learning stage and interests. They offer tailored experiences, develop inclusive strategies, and generate adaptive tests that adjust question difficulty in real time (Kasneci et al., 2023; Xu et al., 2024).

Among these applications, automated question generation stands out as a transformative tool. It can reduce educators' manual workload by enabling rapid development of sophisticated assessment instruments and by enabling the dynamic creation of diverse item variations, including plausible distractor answers, tailored to specific learning objectives (Hang et al., 2024; Kasneci et al., 2023; Maity et al., 2023; Tran et al., 2023; Xu et al., 2024).

In this context, Multiple-Choice Questions (MCQs) are widely used in formative, diagnostic, and summative assessments because they enable objective grading, large-scale administration, and coverage of diverse topics within a domain. Despite these advantages, designing good MCQs requires more than writing grammatically correct prompts. An adequate question must present conceptual clarity, alignment with the content, a level of complexity compatible with the educational objective, only one correct alternative, and plausible distractors (Case & Swanson, 2016; Haladyna et al., 2002).

Regarding MCQs, the automatic generation involves both linguistic and pedagogical aspects, and prompt engineering plays a central role, since the way the task is presented to the model can directly influence the output format, domain adherence, question diversity, level of complexity, and clarity of the alternatives (Maity et al., 2023; White et al., 2023). Strategies such as *zero-shot*, *few-shot*, *chain-of-thought*, *exemplar-guided*, and *template-based* offer different ways of guiding generation, ranging from direct instructions to the use of examples, intermediate reasoning, or fixed output structures (Brown et al., 2020; Wei et al., 2022).

Recent literature has investigated MCQ generation with LLMs in different contexts, such as Computing, Engineering, Medicine, and basic education (Al Shuraiqi et al., 2026; Biancini et al., 2024; Shu et al., 2025; Tran et al., 2023). These studies report important advances in the automated production of assessment items but also indicate that question quality cannot be evaluated solely by the textual appearance of the generated output. Criteria such as distractor quality, content adherence, pedagogical relevance, and cognitive level require careful analysis and, in many cases, human validation (Scaria et al., 2024a, 2024b).

In this work, we extend our previous study (Quincozes et al., 2025) by conducting a new experimental round in 2026 on the automated generation of MCQs by LLMs. In particular, we maintain the same domain, Software Requirements Engineering, the five Prompt Engineering techniques, the evaluation criteria, and the professor validation adopted in the original protocol. Since the LLM ecosystem evolves rapidly, with models being updated, replaced, or discontinued, the comparison between the 2025 and 2026 rounds does not aim to establish direct equivalence

between models. Instead, it seeks to observe the stability and variation of results under a similar experimental protocol. The research questions that guide this study are presented below.

- **Central Research Question (RQ):** To what extent are the results obtained in the automated generation and evaluation of MCQs by LLMs maintained or changed in a new experimental round with models available in 2026?
- **RQ1:** Which Prompt Engineering techniques show the best performance in 2026?
- **RQ2:** Which patterns observed in 2025 remain or change in 2026?
- **RQ3:** To what extent does LLM-based evaluation converge with or diverge from professor evaluation?
- **RQ4:** Which evaluation criteria remain most fragile in the automated generation of MCQs?

The main contributions of this article are: (i) a new experimental round with models available in 2026; (ii) the expansion of the set of generated questions in relation to the original study; (iii) the comparison between automatic evaluation by LLMs and professor evaluation; (iv) the analysis of error patterns identified by expert professors; and (v) the discussion on stability and reproducibility in educational studies that use LLMs.

From a practical perspective, the study contributes to teachers by providing evidence on which prompting strategies tend to produce questions that are more positively evaluated by expert professors, as well as by identifying criteria that still require careful human review, such as relevance, diversity, and complexity. From a research perspective, the study contributes by comparing two experimental rounds conducted under a similar protocol, supporting the discussion on stability, reproducibility, and model-dependence in empirical studies with LLMs. For the field of Computers in Education, the work contributes evidence on the responsible use of LLMs as assistive tools for the generation and initial screening of assessment items, while reinforcing that pedagogical validation remains necessary.

The remainder of this article is organized as follows. Section 2 presents the theoretical background on LLMs in education, multiple-choice questions, Prompt Engineering, evaluation criteria, and reproducibility challenges. Section 3 discusses related work and positions this study in relation to recent literature. Section 4 describes the experimental design, the 2025 and 2026 rounds, the models used, and the automatic and professor evaluation procedures. Section 5 presents and discusses the results, including the comparison between experimental rounds. Finally, Section 6 presents the conclusions, limitations, and future work.

2 Background

In this section, we outline the key theoretical foundations that underpin the methodological design of this study. Instead of offering a general overview of Large Language Models (LLMs), we focus on the components needed to understand automated generation of multiple-choice questions (MCQs) in educational settings. This includes an exploration of the use of LLMs in education, the pedagogical significance of MCQs, the importance of Prompt Engineering, evaluation criteria, and the challenges of reproducibility in empirical studies involving LLMs.

2.1 LLMs in Education and Assessment Support

LLMs have been widely investigated in educational contexts due to their ability to understand, generate, and transform natural language texts. In the education field, these models have been explored in tasks such as instructional content generation, feedback production, writing support, intelligent tutoring, automated grading, material personalization, and assessment support (Cho et al., 2024; Kasneci et al., 2023; Xu et al., 2024). In particular, the ability to generate questions, explanations, and contextualized examples makes LLMs relevant candidates for supporting teaching practices that require time and effort, such as the construction of assessment instruments.

Recent literature indicates that one of the main opportunities for using LLMs in education lies in scalability. These models can support rapid production of educational materials, enabling variations in questions, examples, and explanations for different learning objectives. In addition, these models can contribute to personalized teaching by adapting content to the student's level, task style, or addressed domain. In assessment scenarios, this capability is especially relevant, as it may reduce part of the operational workload associated with the initial development of questions and allow professors to focus their efforts on curating, validating, and pedagogically refining the generated items (Kasneci et al., 2023; Xu et al., 2024).

However, the use of LLMs in educational processes also presents important limitations. These models may produce inconsistent responses, present factually incorrect information, reinforce biases present in training data, or generate content that is linguistically plausible but pedagogically inadequate (Huang et al., 2025; Kasneci et al., 2023). In the context of assessment, such limitations are particularly sensitive, since an apparently well-written question may contain ambiguity, weak distractors, low adherence to the educational objective, or an inadequate level of complexity. Thus, the textual quality of an output generated by an LLM should not be confused with pedagogical quality (Huang et al., 2025).

Therefore, this work adopts an assistive perspective on the use of LLMs in education. The models are intended to support the generation and initial screening of questions, not as substitutes for a professor's judgment. This perspective aligns with the literature, which recommends human supervision in educational applications of AI, especially in assessment tasks, where validity, clarity, content alignment, and adequacy to the cognitive level are fundamental (Cho et al., 2024; Kasneci et al., 2023). For this reason, the methodology of this study combines automatic evaluation by LLMs with human evaluation by expert professors, allowing comparison of automated judgments with the pedagogical interpretations of human evaluators.

2.2 Multiple-Choice Questions (MCQs)

MCQs are assessment instruments generally composed of a stem, a set of alternatives, one correct answer, and incorrect alternatives called distractors. The quality of an MCQ does not depend only on the grammatical correctness of the text, but also on the articulation among the stem, alternatives, learning objective, and the level of reasoning required from the student (Case & Swanson, 2016; Haladyna et al., 2002). For this reason, the automatic generation of MCQs represents a more complex educational task than simple text generation.

MCQs are widely used in the educational field because they enable objective assessment, large-scale application, and efficient grading. They also allow coverage of different topics within a

knowledge domain and can be used in diagnostic, formative, or summative assessments. However, developing good MCQs requires methodological care. Classical guidelines for item construction recommend that the question be aligned with learning objectives, present a clear stem, contain only one correct answer, avoid ambiguity, reduce superficial clues, and use plausible distractors (Case & Swanson, 2016; Haladyna et al., 2002).

Distractors play a central role in the quality of a multiple-choice question. They should not be obviously incorrect alternatives, as this reduces the item's discriminatory capacity. Instead, good distractors should represent plausible errors, misconceptions, or incomplete interpretations that may attract students who have not yet adequately mastered the content (Haladyna et al., 2002). When distractors are weak or artificial, students may identify the correct answer by superficial elimination, without mobilizing the expected knowledge. Thus, a question may be grammatically fluent and still have low assessment quality.

Another important aspect is the cognitive complexity level of the question. MCQs can assess everything from factual knowledge to interpretation, application, analysis, and evaluation. The revised Bloom's Taxonomy proposes an organization of cognitive processes into levels such as remembering, understanding, applying, analyzing, evaluating, and creating (Anderson & Krathwohl, 2001).

While not every MCQ needs to assess higher levels of learning, higher-quality questions typically require students to go beyond simple memorization. They often involve interpreting situations, relating concepts, or applying principles to a specific context.

This observation is particularly relevant to this study, as the results analyzed consider various quality criteria. A question can demonstrate high grammatical fluency but still lack diversity or complexity. Therefore, evaluating MCQs generated by language models requires considering multiple dimensions. The distinction between textual and pedagogical quality supports the need for criteria such as diversity, complexity, and relevance, in addition to fluency and answerability.

2.3 Prompt Engineering for Question Generation

Prompt Engineering consists of the process of formulating, structuring, and adjusting natural language instructions to guide the behavior of LLMs in a given task. Instead of modifying the internal parameters of the model, Prompt Engineering acts on the input provided to the system, influencing the format, content, level of detail, and type of reasoning expected in the output (White et al., 2023). In educational tasks, this practice is relevant because small changes in the instruction may significantly alter the quality, structure, and pedagogical adequacy of the generated material.

In the context of MCQ generation, prompts can specify the question domain, the number of alternatives, the expected format, the presence of a single correct answer, the difficulty level, the type of content addressed, and even the nature of the distractors. Thus, Prompt Engineering works as a mechanism of partial control over generation, allowing different instructional strategies to be compared without the need for model retraining or fine-tuning. To investigate these effects, we describe below the main prompting methods established in recent literature, emphasizing those selected for our comparative analysis.

Zero-shot prompting consists of providing the model only with the task description, without additional examples. This approach is simple, low-cost, and useful for evaluating the model's

ability to perform a task when given direct instructions. In LLMs, *zero-shot* capability is associated with the model's generalization to tasks not accompanied by explicit demonstrations (Brown et al., 2020). In the case of MCQ generation, a *zero-shot* prompt directly requests the creation of a question within a specified domain and format.

Few-shot prompting includes examples within the prompt to guide the model on the expected input and output patterns. This technique explores the in-context learning capability of LLMs, in which demonstrations provided in the prompt condition the subsequent response (Brown et al., 2020). In question generation, examples can help the model reproduce a desired structure, maintain consistency in the format of the alternatives, and approximate the output to an expected educational standard.

Chain-of-thought prompting encourages the model to perform intermediate reasoning steps before producing the final answer. This technique was proposed to improve the performance of LLMs in tasks that require complex reasoning, especially when combined with larger-scale models (Wei et al., 2022). In MCQ generation, this strategy may encourage the model to plan the assessed concept, the required reasoning, and the distinction between the correct alternative and the distractors. However, its effectiveness may vary across models, domains, and task formats, especially when the final goal requires a concise, well-structured item.

Exemplar-guided prompting is a variation based on the use of examples aligned with the task domain. Unlike generic examples, the provided example aims to reflect the content, language, and structure expected in the specific domain. In this study, this means guiding generation based on examples compatible with Requirements Engineering. This approach may favor greater semantic and pedagogical adherence, as it brings the model closer to the expected standard for the evaluated context.

Finally, *template-based prompting* uses a fixed structure that the model completes. This technique aims to improve control over the output format, reducing undesired variations and promoting standardization. In MCQ generation, a template can explicitly define fields such as the stem, alternative A, alternative B, alternative C, and alternative D, as well as the correct answer. This standardization facilitates automatic data extraction and comparison across models, although it may limit the structural diversity of the generated questions.

The literature on Prompt Engineering indicates that there is no universally superior technique for all tasks and models. The effectiveness of each strategy may depend on the model architecture, size, training, domain, level of prompt detail, and the type of expected output (Wei et al., 2022; White et al., 2023). Therefore, a systematic comparison of techniques is necessary to understand how different instructional approaches influence the quality of generated MCQs.

2.4 Evaluation Criteria for Generated Questions

The evaluation of automatically generated questions requires criteria that capture the different dimensions of item quality. In this study, the five criteria used in the original study were maintained: grammatical fluency, answerability, diversity, complexity, and relevance (Quincozes et al., 2025). These criteria are also aligned with the evaluation approach adopted in studies on automatic MCQ generation with LLMs, such as MCQGen, which considers linguistic, structural, and educational dimensions in the analysis of generated questions (Hang et al., 2024).

Grammatical fluency evaluates the linguistic quality of the question, considering clarity, cohesion, grammatical correctness, and textual structure. A question with good fluency should present a comprehensible stem, well-written alternatives, and the absence of errors that hinder interpretation. This criterion is important because linguistic problems may compromise the validity of the item, confusing the student or diverting attention from the assessed content (Chico et al., 2025; Hang et al., 2024; Scaria et al., 2024b).

Answerability verifies whether the question fulfills the proposed task. In the case of MCQs, this means presenting a coherent question or assessment situation, with alternatives compatible with the stem and a single, clearly correct answer. This criterion concerns the absence of ambiguity and the item's internal consistency. A question may be fluent but fail in answerability if two alternatives are equally defensible, if no alternative adequately answers the stem, or if the correct answer is not well defined (Hang et al., 2024; Scaria et al., 2024b).

Diversity considers the thematic, structural, and stylistic variation of the generated questions. In a set of MCQs, diversity means avoiding excessive repetition of topics, formulations, stem patterns, or types of alternatives. This criterion is relevant because automatic generation systems may produce questions that are very similar to one another, especially when rigid prompts are used or when the model relies on frequent patterns. Low diversity reduces the pedagogical usefulness of the generated set by limiting domain coverage and potentially making the assessment predictable (Chico et al., 2025).

Complexity evaluates the level of reasoning required to correctly answer the question. This criterion can be related to the revised Bloom's Taxonomy, as it distinguishes questions that require only the recall of definitions from those that demand understanding, application, analysis, or evaluation (Anderson & Krathwohl, 2001). In the context of this study, a question of higher complexity is not necessarily more difficult in statistical terms, but rather one that requires greater cognitive elaboration to be solved.

Relevance evaluates the alignment of the question with the Requirements Engineering domain and the expected educational objectives. A relevant question should address concepts, practices, or problems pertinent to the domain, avoiding generic formulations or content that is tangential to the subject matter. This criterion is especially important in technical areas, since a question may appear formally adequate but fail to assess the discipline's core knowledge.

Although these five criteria are useful for evaluating different dimensions of generated MCQs, they do not capture all possible aspects of quality. The analysis of distractors, for example, may require specific criteria regarding plausibility, homogeneity, independence, and discriminatory power of the alternatives (Case & Swanson, 2016; Haladyna et al., 2002). In addition, the psychometric difficulty of a question and its ability to discriminate among students with different proficiency levels can be assessed more robustly using application data from students. Thus, the criteria used in this study provide an initial and comparative evaluation of question quality, but they may be complemented by qualitative and psychometric analyses in future work.

2.5 Reproducibility and Instability in Studies with LLMs

Empirical studies with LLMs present specific reproducibility challenges. Unlike traditional systems, LLMs may produce different outputs for the same input depending on factors such as model

version, inference parameters, generation randomness, token limit, prompt format, and the API's policies (Angermeir et al., 2025; Baltes et al., 2025). In addition, models made available by external providers may be updated, replaced, renamed, or discontinued, which makes it difficult to exactly reproduce experiments conducted on previous dates.

Recent literature on empirical studies involving LLMs recommends that researchers explicitly report the model's role in the study, the versions used, configuration parameters, prompts, interaction logs, evaluation metrics, and the limitations associated with these models (Baltes et al., 2025). Human validation of outputs generated by LLMs is also recommended when these outputs affect qualitative interpretations or domain-related decisions. These recommendations are particularly relevant in educational studies, in which quality assessment involves pedagogical judgment.

3 Related Work

The automated generation of MCQ with the support of LLMs has received increasing attention in recent literature. This interest is related to the potential of these models to support professors in producing assessment instruments, reduce the initial effort required to create questions, and enable greater scale in the development of educational materials. However, MCQ generation involves challenges beyond text production, as it requires content alignment, conceptual clarity, a single correct alternative, plausible distractors, and an appropriate cognitive level.

Tran et al. (2023) investigated the use of LLMs for MCQ generation in Computing courses, providing an important reference for the use of these models in technical disciplines. Complementarily, Maity et al. (2023) explored *prompting*-based techniques for educational question generation, including human evaluation of the produced items. These works indicate that LLMs can support question development, but they also show that item quality depends on how the task is presented to the model and on the subsequent validation of the generated outputs.

Other recent studies further examine the relationship between LLMs, *prompting*, and the pedagogical quality of questions. Li and Zhang (2024) proposed an LLM-guided approach for controllable question generation, based on the idea of planning before formulating the question. Scaria et al. (2024a) investigated the generation of educational questions at different levels of Bloom's Taxonomy, while Scaria et al. (2024b) analyzed the quality of LLM-generated questions at different cognitive levels. These studies are relevant because they relate automatic question generation to educational aspects, such as complexity, adherence to cognitive level, and pedagogical quality.

The literature has also advanced in the automatic evaluation of MCQs generated by LLMs. Hang et al. (2024) presented MCQGen, an LLM-based system for generating MCQs aimed at personalized learning. Mucciaccia et al. (2025) investigated LLM-based systems for automatic MCQ generation and evaluation in a case study involving university resolutions. Yao et al. (2025) proposed an iterative process of generating, critiquing, correcting, and comparing MCQs, showing a recent tendency to use LLMs not only to generate questions, but also to evaluate and refine them.

Human validation remains a recurring concern in these studies. Biancini et al. (2024) presented a methodology for MCQ generation with LLMs and analyzed the results based on educa-

tors' perceptions. Shu et al. (2025) explored MCQ generation with multimodal models in higher education in Engineering, highlighting the alignment of questions with learning objectives and the need for human supervision. In the medical context, Al Shuraiqi et al. (2026) investigated Prompt Engineering strategies for generating case-based MCQs, using multiple models and expert validation. These works bring automatic question generation closer to pedagogical criteria, such as relevance, difficulty, distractor quality, and adequacy to the educational context.

In addition to complete MCQ generation, some works investigate specific components of question quality. Bitew et al. (2023), for example, analyzed distractor generation with LLMs and *predictive prompting*, an aspect directly related to the plausibility of incorrect alternatives. Elkins et al. (2024), in turn, discussed the use of LLMs and Bloom's Taxonomy to support professors in creating educational quizzes. Although these studies do not have exactly the same focus as this work, they contribute to the discussion on pedagogical quality, cognitive complexity, and professor participation in the educational use of LLMs.

Table 1 presents a comparison between this study and related works across five aspects: the use or analysis of Prompt Engineering, the use of multiple language models, automated evaluation by LLMs, participation of human evaluators, and focus on MCQs. The table is not intended to rank the studies, but to highlight how recent literature has explored different combinations of these elements.

Table 1: Comparison between this study and related works.

Reference	PE	LLMs	LLM Eval.	Hum. Eval.	MCQs
Quincozes et al., 2025	✓	✓	✓	✓	✓
Al Shuraiqi et al., 2026	✓	✓	×	✓	✓
Biancini et al., 2024	✓	✓	×	✓	✓
Shu et al., 2025	✓	✓	×	✓	✓
Mucciaccia et al., 2025	✓	×	✓	✓	✓
Bhowmick et al., 2023	✓	✓	×	✓	✓
Tran et al., 2023	×	✓	×	✓	✓
Hang et al., 2024	×	✓	✓	×	✓
Li and Zhang, 2024	✓	×	×	✓	✓
Scaria et al., 2024a	✓	✓	✓	✓	✓
Scaria et al., 2024b	✓	✓	×	✓	✓
Yao et al., 2025	✓	×	✓	✓	✓

Note: PE = Prompt Engineering/*prompting*; LLMs = use or comparison of multiple language models; LLM Eval. = automated evaluation using LLMs; Hum. Eval. = human/expert evaluation; MCQs = focus on multiple-choice question generation.

Overall, the related work shows that MCQ generation with LLMs is a recent and growing field. The literature has advanced in different directions, including the use of prompts to control generation, model comparison, automatic evaluation, human validation, and the analysis of pedagogical criteria. In this scenario, the present work differs by integrating these elements into the same experimental protocol: it compares five Prompt Engineering techniques, uses multiple

generator models, employs cross-evaluation among LLMs, includes professor validation, and analyzes the results in the specific domain of Requirements Engineering. In addition, this version extends the study published in 2025 (Quincozes et al., 2025) by incorporating a new experimental round with models available in 2026, allowing the stability of the findings to be discussed in a context of rapid LLM evolution.

3.1 Synthesis and Research Gap

The related work shows that MCQ generation with LLMs has advanced in several directions, including prompting strategies, model comparison, automatic evaluation, human validation, and the analysis of pedagogical criteria. However, most studies still analyze these dimensions separately or within a single experimental round. As a result, there is limited evidence on whether the observed performance of prompting strategies remains stable when the available model set changes.

A relevant research gap therefore concerns the robustness of experimental protocols over time. In the current LLM ecosystem, models may be updated, replaced, renamed, or discontinued, and API behavior may change across periods. This affects the reproducibility of empirical studies and makes it difficult to determine whether the superiority of a given prompting technique is stable or contingent on a specific set of models.

This study addresses this gap by comparing experimental rounds conducted in 2025 and 2026 under a similar protocol. Instead of treating the comparison as a longitudinal evaluation of the same models, the study examines how evaluation patterns, prompting strategies, and criteria behave across two model-availability scenarios. This framing supports a more realistic discussion about the use of LLMs in educational assessment, emphasizing documentation, human validation, and careful interpretation of results.

4 Methodology

This study was structured as a comparative analysis across two experimental rounds in 2025 and 2026, aimed at examining the performance of various language models in the automated generation of multiple-choice questions in the Requirements Engineering domain. To facilitate a fair comparison, we maintained consistency across several factors, including the application domain, question format, Prompt Engineering techniques, evaluation criteria, evaluation scale, generation and evaluation code, and the professors serving as evaluators. The primary goal was to assess the stability of the results under different model availability scenarios, without assuming direct equivalence among the models used in each round.

The decision to implement two experimental rounds stems from the dynamic nature of Large Language Models (LLMs), which can change across versions, API availability, and execution conditions over time. Consequently, our comparison does not aim to track the longitudinal evolution of the same models; instead, it focuses on how a consistent experimental protocol performs when applied to a new array of available models. This methodological caution is essential in empirical research involving LLMs, as results may be influenced by factors such as model architecture, inference parameters, access policies, and the specific versions used.

4.1 Experimental Design

The experimental design comprised two rounds. In both rounds, five Prompt Engineering techniques applied to the generation of multiple-choice questions were evaluated: *zero-shot*, *few-shot*, *chain-of-thought*, *exemplar-guided*, and *template-based*. These techniques were chosen because they represent recurring strategies for interacting with LLMs in text-generation tasks and enable investigation of how different forms of instruction influence the quality of generated questions.

In the 2025 round, four language models were used: Qwen, LLaMA, Gemma, and DeepSeek. Specifically, the models employed were *qwen-qwq-32b*, *LLaMA-3.1-8b-instant*, *gemma2-9b-it*, and *deepseek-r1-distill-LLaMA-70b*. Each model generated five questions for each of the five prompt techniques, totaling 100 questions.

In the 2026 round, the same experimental protocol was maintained, but with models available in the execution environment at the time of the new data collection: *openai/gpt-oss-120b*, *qwen/qwen3-32b*, *groq/compound*, and *llama-3.1-8b-instant*. To increase the robustness of the analysis and reduce the impact of occasional variations in individual generations, ten questions were generated for each model-technique combination. Thus, the 2026 round produced 200 questions in total, not 200 questions per model. This total results from four generator models, five Prompt Engineering techniques, and ten questions per model-technique combination: $4 \times 5 \times 10 = 200$ questions.

Table 2 presents the general configuration of the two experimental rounds.

Table 2: General configuration of the experimental rounds.

Element	2025 Round	2026 Round
Domain	Requirements Engineering	Requirements Engineering
Question format	Multiple choice	Multiple choice
Alternatives per question	4 alternatives and 1 correct answer	4 alternatives and 1 correct answer
Prompt techniques	5 techniques	5 techniques
Models	Qwen, LLaMA, Gemma, and DeepSeek	GPT-OSS, Qwen3, Compound, and LLaMA
Questions per model-technique combination	5	10
Total number of generated questions	100 in total	200 in total
Evaluation criteria	5 criteria	5 criteria
Evaluation scale	1-to-5 Likert scale	1-to-5 Likert scale
Automatic evaluation	Cross-evaluation by LLMs	Cross-evaluation by LLMs
Human evaluation	4 expert professors	4 expert professors

4.2 Generation of Multiple-Choice Questions

The questions were generated in the Requirements Engineering domain, a central area in Software Engineering education that involves activities such as requirements elicitation, analysis, specification, validation, and management. Each question was required to contain a stem, four alternatives identified by the letters A, B, C, and D, and only one correct alternative.

The five Prompt Engineering techniques used were maintained in both experimental rounds. In *zero-shot prompting*, the model received only the direct instruction to generate a multiple-choice question about Requirements Engineering, without additional examples. In *few-shot prompting*, previous examples of questions were provided to guide the output structure. In *chain-of-thought prompting*, the model was instructed to consider reasoning steps before generating the question. In *exemplar-guided prompting*, an example aligned with the Requirements Engineering domain was presented to induce greater semantic and pedagogical alignment. Finally, in *template-based prompting*, the model received a fixed question structure to be completed.

The adoption of these techniques enabled comparisons of instructional approaches across different levels of task explicitness, example use, and structural control of the output. All questions were generated through the Groq API, with automation implemented in Python. The same codebase was used in both experimental rounds, with changes only to the list of models employed and to the number of questions generated per model-technique combination.

4.3 Evaluation Criteria

The generated questions were evaluated based on five criteria: grammatical fluency, answerability, diversity, complexity, and relevance. These criteria were adopted from the evaluation framework proposed by Hang et al. (2024), as they cover linguistic, structural, and educational dimensions relevant to the analysis of questions automatically generated by LLMs.

Each criterion was evaluated on a 1-to-5 Likert scale, where 1 indicates low adequacy and 5 indicates high adequacy. The same scale was used for both automatic evaluation by LLMs and expert professor evaluation, allowing a direct comparison between the two types of judgment.

Grammatical fluency evaluates the clarity, cohesion, and linguistic correctness of the stem and alternatives. Answerability assesses whether the question adequately addresses the proposed task, presents a single correct alternative, and avoids ambiguity among the options. Diversity considers the thematic, structural, and stylistic variation of the questions, observing whether there is excessive repetition of patterns. Complexity evaluates the level of reasoning required to answer the question correctly, distinguishing merely factual questions from those that require interpretation or analysis. Finally, relevance verifies that the question aligns with the Requirements Engineering domain.

Table 3: Evaluation criteria used in the experimental rounds.

Criterion	Description
Grammatical fluency	Evaluates the clarity, cohesion, linguistic correctness, and textual structure of the question.
Answerability	Evaluates whether the question fulfills the proposed task, with a single correct alternative and no evident ambiguity.
Diversity	Evaluates the thematic, structural, and stylistic variation of the generated questions.
Complexity	Evaluates the level of reasoning required to correctly answer the question.
Relevance	Evaluates the alignment of the question with the Requirements Engineering domain.

4.4 Cross-Evaluation by LLMs

The automatic evaluation was conducted through cross-evaluation among LLMs. In this procedure, each model acted as an evaluator of the questions generated by the other models, preventing a model from evaluating its own outputs. This strategy was adopted to reduce the risk of direct self-evaluation and to allow the analysis of LLMs also as automatic evaluators of educational artifacts.

In each evaluation, the evaluator model received the generated question and assigned scores from 1 to 5 for the five defined criteria: grammatical fluency, answerability, diversity, complexity, and relevance. The responses from the automatic evaluators were requested in a structured format, containing only the numerical scores for each criterion, without textual justifications. This standardization facilitated the consolidation of the results and reduced ambiguities in score extraction.

In the 2025 round, each of the 100 questions was evaluated by the models that had not participated in its generation. In the 2026 round, the same procedure was applied to the 200 generated questions. In both rounds, each automatic evaluation produced five scores, one for each evaluated criterion.

Cross-evaluation by LLMs was used as an automatic measure of quality, but not as a gold standard. The results obtained through this procedure were subsequently compared with professor evaluation, since the pedagogical quality of educational questions involves interpretive criteria that may not be fully captured by language models.

4.5 Professor Evaluation

In addition to automatic evaluation, a human evaluation was conducted by four expert professors. The same professors participated in both experimental rounds, which enhances the comparability of the results between 2025 and 2026. The evaluators have experience in Computing, Software Engineering, and/or Requirements Engineering, and have worked in courses related to these areas.

Each professor evaluated the questions using the same five criteria as in the automatic evaluation, on a 1-to-5 Likert scale. The adoption of the same rubric for LLMs and professors enabled comparison of automated and human judgments across the same quality dimensions.

The participation of multiple human evaluators follows the principle of methodological triangulation proposed by Denzin (1978), which holds that the use of different observers reduces individual biases and strengthens the validity of interpretations. In this study, professor evaluation was used as a pedagogical reference to analyze the extent to which the scores assigned by LLMs approximate or diverge from specialized human judgment.

4.6 Data Consolidation and Analysis

The data from the two experimental rounds were consolidated into structured spreadsheets containing, for each evaluation, the question identifier, the generator model, the prompt technique used, the evaluator, the evaluator type, and the scores assigned to the five criteria. Based on these data, averages were calculated by prompt technique, generator model, evaluation criterion, and evaluator type.

The analysis was organized into three levels. The first level considered the internal results of each experimental round, allowing the identification of which prompt techniques and models achieved the best performance in each year. The second level compared the results of the two rounds, aiming to verify which patterns were maintained and which changed between 2025 and 2026. The third level compared automatic evaluation by LLMs with professor evaluation to identify convergences and divergences between the two.

For the comparison between rounds, the difference between the 2026 and 2025 averages was calculated, as shown in Equation 1.

$$\Delta = \bar{x}_{2026} - \bar{x}_{2025} \quad (1)$$

Where $\bar{x}_{2026}$ represents the average obtained in the 2026 experimental round and $\bar{x}_{2025}$ represents the average obtained in the 2025 experimental round. Positive values of Δ indicate an increase in the average in 2026, negative values indicate a decrease, and values close to zero indicate stability between rounds.

The comparisons were conducted by prompt technique, evaluation criterion, generator model, and evaluator type. Due to model replacements between rounds, the results were not interpreted as a direct comparison of equivalent versions of the same models. Instead, the analysis was treated as a comparison of two experimental scenarios conducted with the same general protocol, allowing the observation of stability and variation in the results across different model sets.

5 Results, Analysis, and Discussion

This section presents the results of the 2026 experimental round and, when relevant, compares them with the findings reported in the 2025 experimental round. The analysis considers the five evaluation criteria defined in the methodology: grammatical fluency, answerability, diversity, complexity, and relevance.

In the 2026 round, 200 multiple-choice questions were generated in the Requirements Engineering domain. Each question was expected to be automatically evaluated by three models that had not participated in its generation, for a total of 600 automatic evaluations. In the consolidated dataset, 599 automatic evaluations were recorded, with one missing evaluation. In addition, the 200 questions were evaluated by four expert professors, totaling 800 professor evaluations. Therefore, all analyses of the 2026 round refer to the full set of 200 generated questions, distributed across the four generator models and the five Prompt Engineering techniques. No individual model generated 200 questions.

In the professor evaluation, all individual criteria were completed in the 800 evaluations. However, the total evaluation column presented 790 valid values, with 10 records lacking a total score. In the set of professor evaluations, 30 incomplete question markings were identified, corresponding to 25 unique questions. To preserve the anonymity of the human evaluators, the four expert professors were identified in this work as P01, P02, P03, and P04.

5.1 Overview of the 2026 Round

Figure 1 presents the general averages obtained in the automatic cross-evaluation and in the professor evaluation for each evaluation criterion.

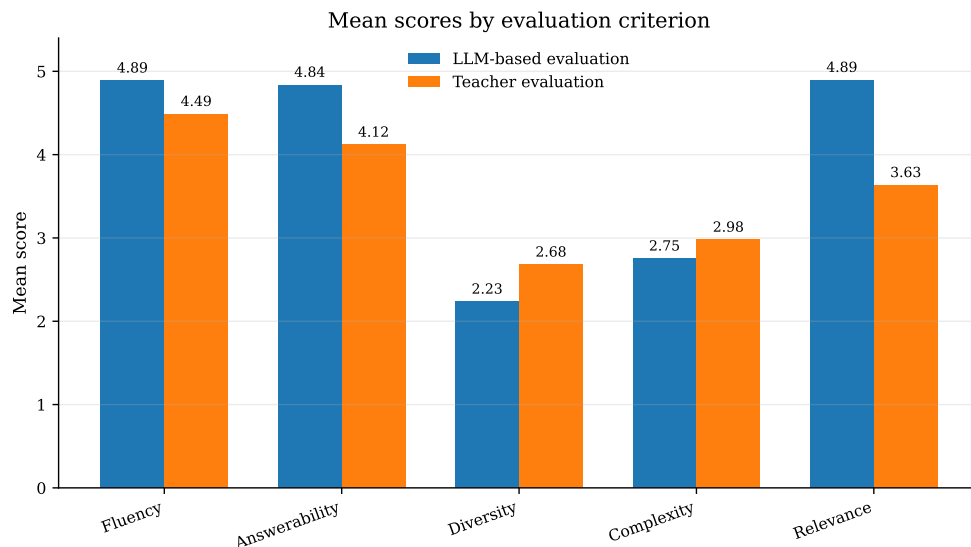


Figure 1: General averages by evaluation criterion in 2026, comparing automatic evaluation by LLMs and professor evaluation.

The results indicate that the LLMs assigned higher averages than the professors in grammatical fluency, answerability, and relevance. In grammatical fluency, the automatic evaluation reached an average of 4.89, while the professor evaluation reached 4.49. In answerability, the automatic average was 4.84, compared to 4.12 in the professor evaluation. The largest difference occurred in relevance, with an average of 4.89 in the automatic evaluation and 3.63 in the professor evaluation.

In contrast, the professors assigned higher averages than the LLMs in diversity and complexity. In diversity, the professor average was 2.68, while the automatic evaluation average was 2.23. In complexity, the professors assigned an average of 2.98, compared to 2.75 in the automatic evaluation. These results suggest that the LLMs tended to evaluate more positively aspects related to textual adequacy, answer format, and apparent domain relevance, whereas the professors were comparatively more attentive to pedagogical and conceptual dimensions.

In both evaluation perspectives, diversity and complexity remained among the weakest criteria. This result indicates that generating questions that are sufficiently varied and cognitively demanding remains a relevant challenge for LLM-assisted MCQ production.

5.2 Prompt Engineering Techniques and Evaluation Criteria

Figure 2 presents the general averages by Prompt Engineering technique, separating the automatic cross-evaluation and the professor evaluation.

In the automatic cross-evaluation, the *chain-of-thought* technique obtained the highest overall average, with 4.13. It was followed by *zero-shot*, with 3.94, *template-based*, with 3.88, *few-*

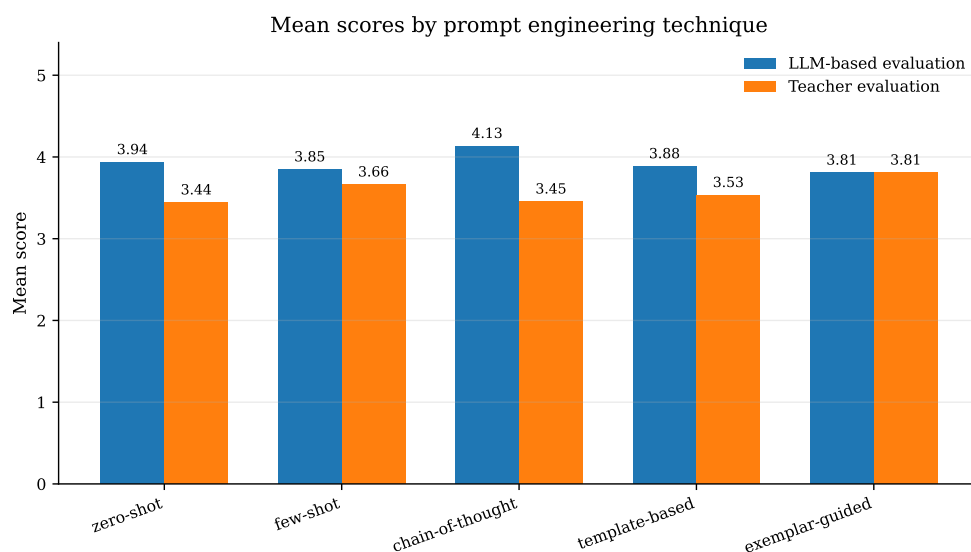


Figure 2: General averages by Prompt Engineering technique in 2026, comparing automatic evaluation by LLMs and professor evaluation.

shot, with 3.85, and *exemplar-guided*, with 3.81. From the perspective of the evaluator LLMs, therefore, the induction of explicit reasoning favored the overall evaluation of the generated questions.

In the professor evaluation, the pattern was different. The technique best evaluated by the professors was *exemplar-guided*, with an average of 3.81, followed by *few-shot*, with 3.66, *template-based*, with 3.53, *chain-of-thought*, with 3.45, and *zero-shot*, with 3.44. Thus, while the LLMs favored *chain-of-thought*, the professors assigned the highest average to the technique based on domain-oriented examples.

Figures 3 and 4 detail this behavior by evaluation criterion, considering professor evaluation and automatic evaluation, respectively.

In the professor evaluation, grammatical fluency presented high averages across all techniques, ranging from 4.38 in *chain-of-thought* to 4.58 in *zero-shot*. In answerability, *template-based* obtained the highest professor average, with 4.24, suggesting that more controlled prompt structures may favor the production of questions with more clearly defined stems, alternatives, and correct answers.

The diversity and complexity criteria presented lower averages in the professor evaluation. In diversity, *exemplar-guided* obtained the highest average, with 3.25, while *zero-shot* obtained the lowest, with 2.24. In complexity, *exemplar-guided* also presented the highest average, with 3.23, followed by *few-shot*, with 3.02, *chain-of-thought*, with 2.99, *template-based*, with 2.90, and *zero-shot*, with 2.75. In relevance, *exemplar-guided* again stood out, with an average of 3.97, while *template-based* presented the lowest average, with 3.49, followed by *chain-of-thought*, with 3.50.

In the automatic evaluation, *chain-of-thought* stood out mainly in diversity and complexity, with averages of 2.74 and 3.29, respectively. These values were higher than those of the other techniques in these two criteria, which helps explain its better overall performance in the

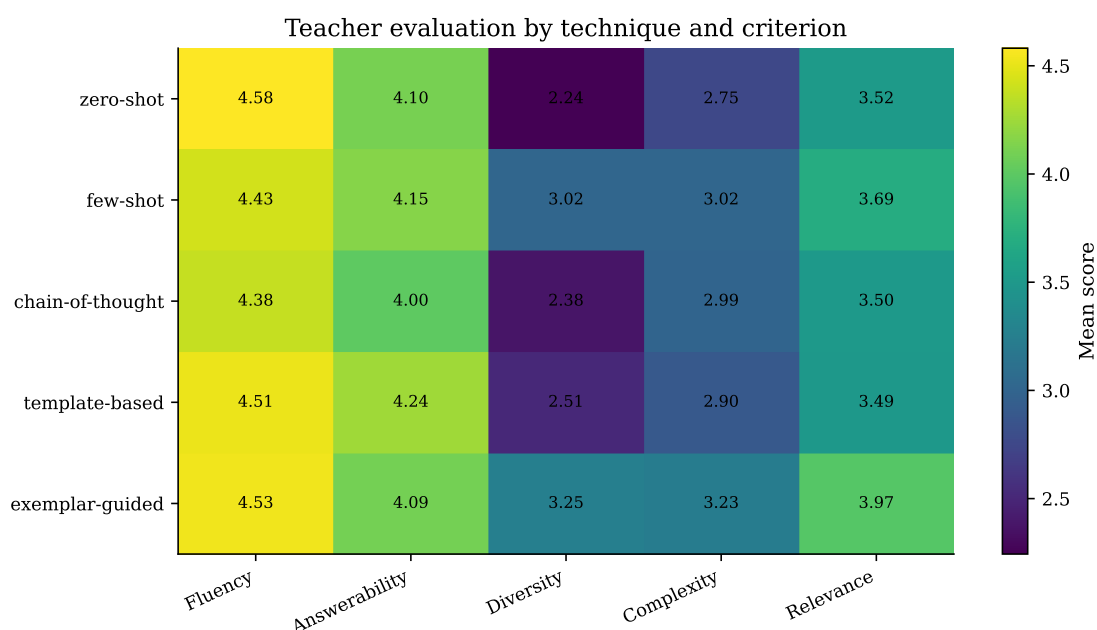


Figure 3: Professor averages by Prompt Engineering technique and evaluation criterion.

cross-evaluation by LLMs. However, this advantage was not repeated in the professor evaluation, indicating that LLMs may have valued the presence of structured reasoning or greater textual elaboration as a sign of quality, while professors evaluated more critically whether this structure effectively resulted in an adequate, clear, and pedagogically useful question.

5.3 Models, Evaluators, and Judgment Divergences

Figure 5 presents the general averages by generator model, considering the automatic evaluation and professor evaluation separately.

In the automatic cross-evaluation, the Compound model obtained the highest overall average, with 4.31. It was followed by Qwen, with 4.02, GPT-OSS, with 3.71, and LLaMA, with 3.65. From the perspective of the evaluator LLMs, Compound and Qwen produced questions that were better evaluated on average.

However, the professor evaluation presented another pattern. Among the professors, GPT-OSS obtained the highest overall average, with 3.91, followed by Qwen, with 3.54, Compound, with 3.44, and LLaMA, with 3.42. Thus, the model best evaluated by the LLMs was not the same model best evaluated by the professors.

This divergence indicates that automatic evaluation and professor evaluation captured distinct dimensions of question quality. Compound presented the highest automatic average, but did not lead the professor evaluation. GPT-OSS, in turn, did not lead the automatic evaluation, but was the model best evaluated by the professors.

Figure 6 presents the difference between the averages assigned by the LLMs and by the professors for each evaluation criterion.

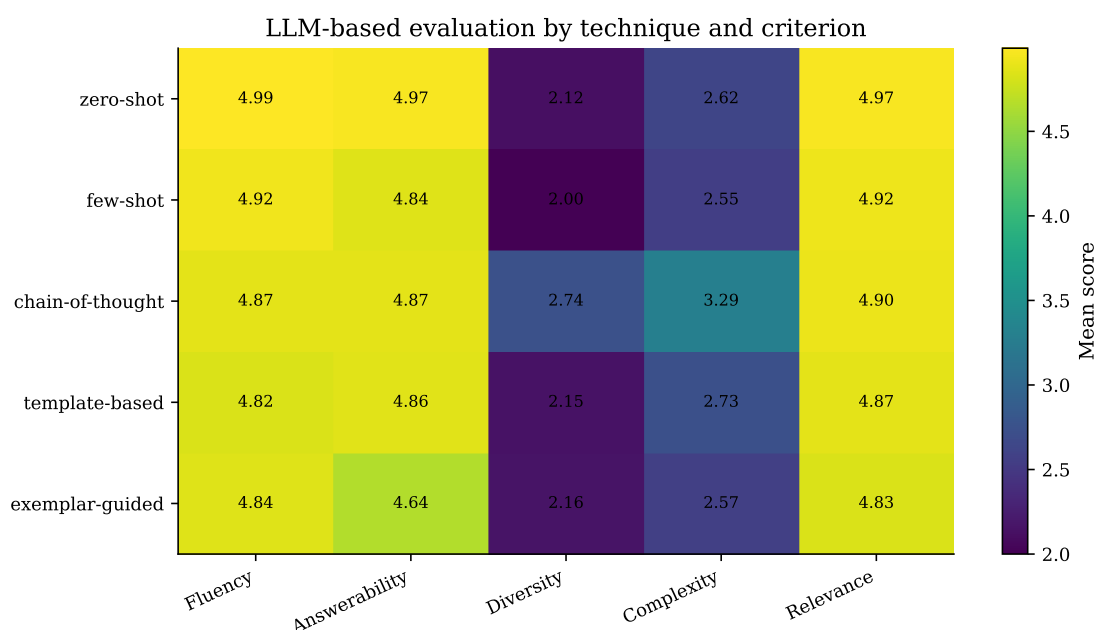


Figure 4: Automatic evaluation averages by Prompt Engineering technique and evaluation criterion.

The largest differences occurred in relevance, answerability, and grammatical fluency. In relevance, the difference was 1.26 points in favor of the LLMs; in answerability, it was 0.72 points; and in grammatical fluency, it was 0.40 points. On the other hand, in diversity and complexity, the professors assigned higher averages than the LLMs, resulting in negative differences of -0.44 and -0.23, respectively.

It is important to note that these differences are interpreted descriptively. No inferential statistical test was applied in this study to determine statistical significance. Therefore, the reported differences should be understood as observed differences in average scores within the analyzed dataset, rather than as statistically significant effects.

Taken together, these results show that automatic evaluation and professor evaluation did not produce equivalent judgments. The LLMs tended to assign higher scores to aspects related to textual adequacy, answerability, and apparent relevance, while the professors were more critical in these criteria and assigned relatively higher averages in diversity and complexity. Thus, automatic evaluation may be useful as a large-scale screening mechanism, but it does not replace human evaluation in more sensitive pedagogical aspects, such as ambiguity, conceptual adequacy, quality of alternatives, item diversity, and level of cognitive complexity.

5.4 Incomplete Questions and Comparison with the 2025 Round

In addition to the scores assigned to the five evaluation criteria, the professors indicated whether each question was incomplete. This marking was used as a qualitative indicator of more serious failures in question generation. It should be noted that incompleteness markings represent individual professor evaluations; therefore, the same question may have received multiple markings when identified as problematic by multiple evaluators.

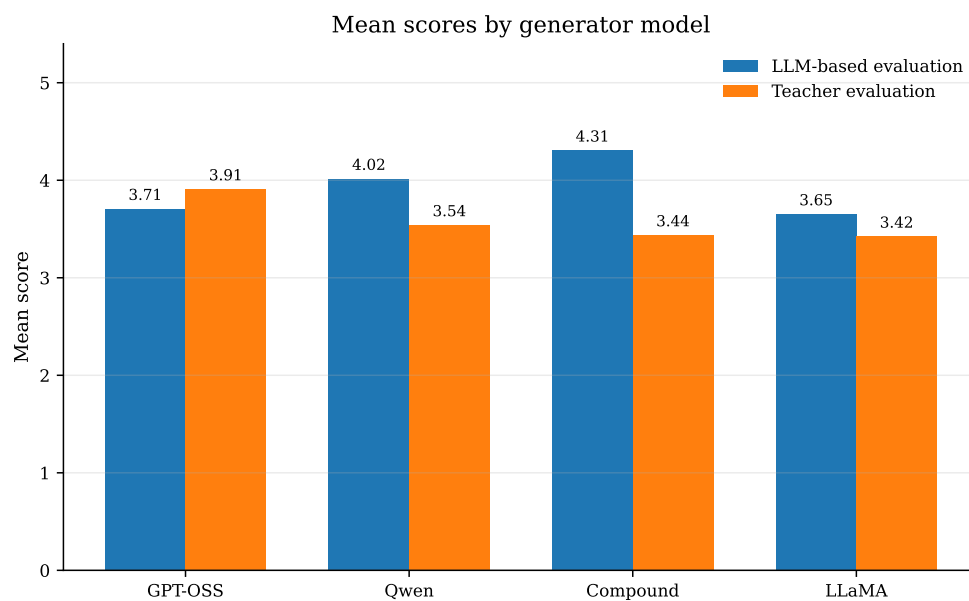


Figure 5: General averages by generator model in 2026, comparing automatic evaluation by LLMs and professor evaluation.

In the set of 800 professor evaluations, 30 incompleteness markings were recorded, corresponding to 25 unique questions. The distribution of these markings among the evaluators was not homogeneous, as presented in Table 4.

Table 4: Number of incompleteness markings by professor evaluator.

Evaluator	Incompleteness markings
P01	2
P02	2
P03	8
P04	18

Question 62 was the most critical case, as it was marked as incomplete by all four professor evaluators. This question was generated by the Qwen model using the *few-shot* technique. The evaluators' observations indicate format and content problems: the presence of internal reasoning in the final answer, excerpts in English, unfinished formulation, and the possibility of more than one correct answer.

Question 86 also presented a strong negative consensus, having been marked as incomplete by three professor evaluators. This question was generated by the Qwen model using the *exemplar-guided* technique. As in Question 62, the problem was associated with the presence of internal reasoning and the absence of a properly structured final question.

Figure 7 presents the distribution of incompleteness markings by generator model and Prompt Engineering technique.

It can be observed that the Qwen model concentrated most of the incompleteness markings: 20 of the 30 occurrences. The highest concentrations occurred in Qwen with the *exemplar-guided*

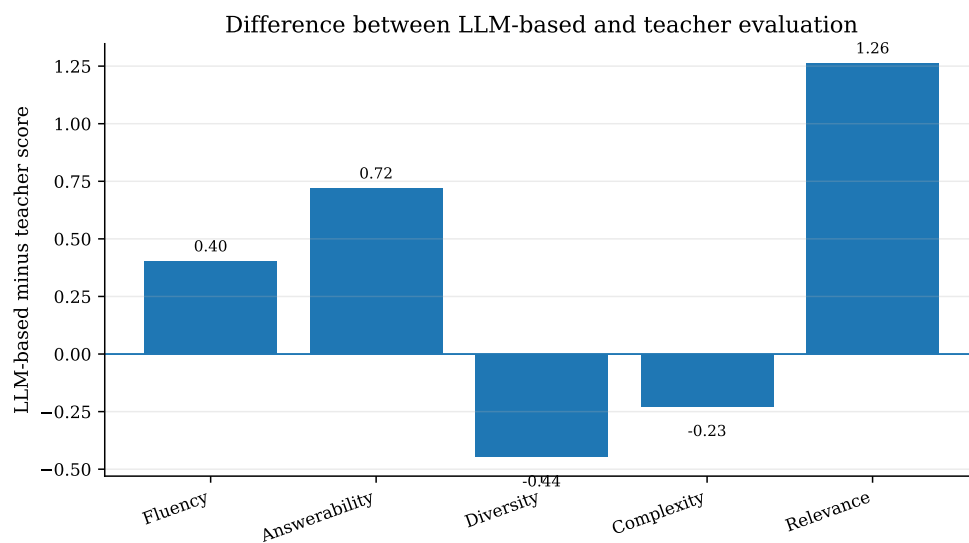


Figure 6: Difference between the averages of automatic evaluation by LLMs and professor evaluation by evaluation criterion.

and *zero-shot* techniques, both with 7 markings, followed by Qwen with *few-shot*, with 5 markings. This result is particularly relevant because Qwen obtained the second-highest average in the automatic cross-evaluation but accounted for most of the serious failures identified by the professors.

This pattern indicates that the problem is not limited to the Prompt technique but also involves the interaction between the technique and the generator model. Therefore, high quantitative averages may coexist with relevant specific problems, especially when there are formatting failures, exposed internal reasoning, or unfinished questions.

The errors identified by the professors can be grouped into three main categories. The first corresponds to LLM output failures, such as the presence of internal reasoning, English excerpts, textual duplication, or unfinished questions. The second involves problems with question quality, such as ambiguity, the possibility of more than one correct alternative, or an answer that is considered debatable. The third refers to conceptual or terminological inconsistencies, such as treating a phase of Requirements Engineering as a technique.

Regarding the 2025 round, the previous study evaluated 100 multiple-choice questions generated by four language models, combining five Prompt Engineering techniques. In that experiment, each model generated 25 questions, with five questions per technique. In the 2026 round, the experimental set was expanded to 200 questions, maintaining the five Prompt Engineering techniques and the Requirements Engineering domain, but using a new set of models available during data collection.

In 2025, the *zero-shot* technique presented the best overall average performance in the previous study. In 2026, this pattern changed: in the automatic cross-evaluation, the highest average was obtained by *chain-of-thought*; in the professor evaluation, the highest average was obtained by *exemplar-guided*. This change indicates that there is no universally superior technique across all scenarios, since performance varies depending on the models used, the type of evaluation, and the prioritized criteria.

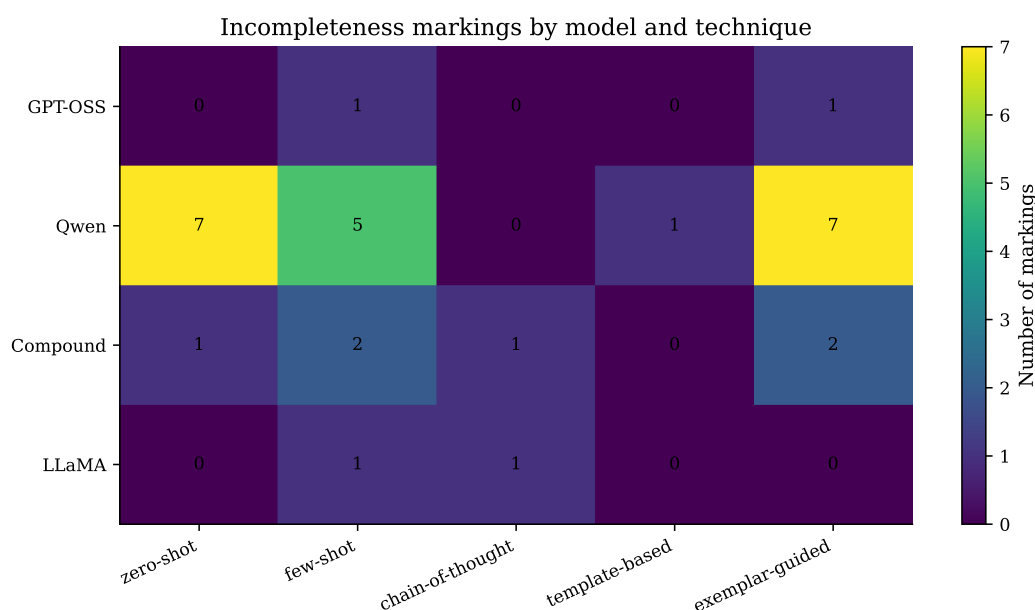


Figure 7: Distribution of incompleteness markings by generator model and Prompt Engineering technique.

Despite this change, some patterns remained consistent. Grammatical fluency continued to be one of the best-evaluated criteria, both in automatic and professor evaluations. On the other hand, diversity and complexity remained the weakest criteria, indicating that generating varied, cognitively more demanding questions remains a challenge for LLMs used for automatic MCQ generation.

Taken together, the results indicate that automatic evaluation by LLMs and professor evaluation capture distinct dimensions of MCQ quality. Although LLMs can support initial large-scale screening, the incompleteness markings and differences in evaluation criteria indicate that professor validation remains essential for identifying pedagogical, conceptual, and structural failures that are not evident in aggregated averages.

5.5 Practical Recommendations for Educators

The results provide practical implications for educators interested in using LLMs to support assessment design. First, LLMs may be useful for generating initial drafts of MCQs, especially when teachers need a starting point for question writing or when multiple variations of an item are required. The high averages observed for grammatical fluency and answerability indicate that LLM-generated questions often present adequate textual structure and a recognizable question-answer format.

Second, teachers should not rely on textual fluency as evidence of pedagogical quality. The lower averages observed for diversity and complexity indicate that generated questions may still be repetitive or cognitively shallow. Therefore, professor review should focus on whether the question is aligned with the learning objective, whether the distractors are plausible, whether there is only one correct answer, and whether the item requires the expected level of reasoning.

Third, the results suggest that domain-oriented examples may be useful when prompting LLMs for assessment design. In the 2026 professor evaluation, exemplar-guided prompting obtained the highest average among the Prompt Engineering techniques. This suggests that teachers may benefit from providing examples that reflect the expected domain, format, and pedagogical level of the questions.

Finally, the use of LLMs for assessment design should be organized as a human-in-the-loop workflow. In this workflow, the LLM generates initial items, automatic evaluation can support large-scale screening, and the professor performs the final validation and refinement. This approach preserves the teacher's pedagogical authority while taking advantage of the scalability of LLM-based generation.

6 Conclusion

This article presented a comparative analysis of LLM-generated multiple-choice questions in the Requirements Engineering domain. The study extends the previous SBIE publication by introducing a new experimental round conducted in 2026, while maintaining the same domain, Prompt Engineering techniques, evaluation criteria, and validation protocol. The 2026 round expanded the experimental set to 200 generated questions and enabled the analysis of models available in a new execution scenario.

The results indicate that automatic evaluation by LLMs and professor evaluation produced different descriptive patterns. In the automatic cross-evaluation, the LLMs assigned an overall average of 3.92, while the professors assigned an overall average of 3.58. The LLMs tended to assign higher scores for grammatical fluency, answerability, and relevance, whereas the professors assigned higher averages for diversity and complexity. These differences should be interpreted descriptively, since no inferential statistical test was applied to determine statistical significance.

The findings also show that the best-performing Prompt Engineering technique depends on the evaluator type. In the automatic evaluation, *chain-of-thought* obtained the highest overall average, with 4.13. In the professor evaluation, *exemplar-guided* obtained the highest average, with 3.81. This suggests that LLM evaluators may favor more elaborate or reasoning-structured outputs, while professors appear to consider more carefully the pedagogical adequacy, conceptual clarity, and usefulness of the generated question as an assessment item.

The analysis by generator model also revealed divergences between automatic and professor evaluations. Compound obtained the highest overall average in the automatic evaluation, whereas GPT-OSS obtained the highest overall average in the professor evaluation. At the same time, Qwen concentrated most of the incompleteness markings identified by the professors. This result reinforces that aggregated averages may hide relevant specific failures, such as exposed internal reasoning, excerpts in English, unfinished questions, ambiguity among alternatives, or conceptual inconsistencies.

Another important finding concerns the criteria of diversity and complexity, which remained among the weakest aspects of automatic MCQ generation. Even when the generated questions presented high grammatical fluency and adequate structure, the results indicate persistent difficulty in producing items that are sufficiently varied and cognitively demanding. This reinforces the

distinction between textual quality and pedagogical quality: a grammatically correct question is not necessarily a good assessment question.

From a scientific perspective, this study contributes by extending the original experiment with a new experimental round and by analyzing the stability of prompting strategies and evaluation criteria across two model-availability scenarios. It also contributes to the discussion on LLM-based evaluation by showing that automatic evaluators and expert professors may prioritize different aspects of MCQ quality. In this sense, the study provides evidence on both the potential and the limitations of using LLMs to support educational assessment design.

From a practical perspective, the results offer guidance for educators interested in using LLMs to support the production of assessment items. LLMs can be useful for generating initial drafts and supporting large-scale screening, especially when fluency and structural completeness are desired. However, criteria such as relevance, diversity, complexity, ambiguity, distractor quality, and conceptual adequacy still require careful professor review. Therefore, LLMs should be used as assistive tools in a human-in-the-loop workflow, rather than as autonomous producers of final assessment items.

As a limitation, the comparison between 2025 and 2026 should not be interpreted as a longitudinal evaluation of the same models, since the sets of LLMs used in the two rounds are not identical. This difference reflects the instability of the LLM ecosystem, in which models may be updated, replaced, or discontinued. Thus, the results should be understood as a comparison between two experimental rounds conducted with a similar protocol, but with models available in different periods. In addition, although professor evaluation provided an important pedagogical reference, the generated items have not yet been applied to students, which limits psychometric analyses of difficulty, discrimination, and actual performance in an educational context.

Future work should expand the qualitative analysis of problematic questions, investigate distractor quality in greater depth, apply the generated MCQs in real classroom settings, and incorporate psychometric metrics. It is also relevant to explore automatic question repair cycles, in which failures identified by LLMs or professors guide a new stage of item revision and refinement. Finally, additional experimental rounds may contribute to a better understanding of the stability of results in a scenario of rapid evolution of language models.

Award-winning Paper Extended

This paper is an extended version of “Engenharia de Prompt para a Geração Automatizada de Questões Assistida por LLMs: Uma Análise Comparativa,” which was recognized as one of the award-winning papers at the 36th Brazilian Symposium on Computers in Education (SBIE 2025). DOI: <https://doi.org/10.5753/sbie.2025.12891>

References

Al Shuraiqi, S., Alzaabi, A., & Aal Abdulsalam, A. K. (2026). Prompt engineering strategies for generating medical case-based MCQs with large language models: A multi-model com-

- parative study. *Machine Learning and Knowledge Extraction*, 8(2), 41. <https://doi.org/10.3390/make8020041> [GS Search].
- Anderson, L. W., & Krathwohl, D. R. (Eds.). (2001). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. Longman.
- Angermeir, F., Amougou, M., Kreitz, M., Bauer, A., Linhuber, M., Fucci, D., Moyón Constante, F., Méndez, D., & Gorschek, T. (2025). Reflections on the reproducibility of commercial LLM performance in empirical software engineering studies. <https://doi.org/10.48550/arXiv.2510.25506> [GS Search].
- Baltes, S., Angermeir, F., Arora, C., Muñoz Barón, M., Chen, C., Böhme, L., Calefato, F., Ernst, N. A., Falessi, D., Fitzgerald, B., Fucci, D., Kalinowski, M., Lambiase, S., Russo, D., Lungu, M., Prechelt, L., Ralph, P., Treude, C., & Wagner, S. (2025). Guidelines for empirical studies in software engineering involving large language models. <https://doi.org/10.48550/arXiv.2508.15503> [GS Search].
- Bhowmick, A., Jagmohan, A., Vempaty, A., Dey, P., Hall, L., Hartman, J., Kokku, R., & Maheshwari, H. (2023). Automating question generation from educational text. <https://doi.org/10.48550/arXiv.2309.15004> [GS Search].
- Biancini, G., Ferrato, A., & Limongelli, C. (2024). Multiple-choice question generation using large language models: Methodology and educator insights. *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization*. <https://doi.org/10.1145/3631700.3665233> [GS Search].
- Bitew, S. K., Deleu, J., Develder, C., & Demeester, T. (2023). Distractor generation for multiple-choice questions with predictive prompting and large language models. <https://doi.org/10.48550/arXiv.2307.16338> [GS Search].
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 1877–1901. [GS Search].
- Case, S. M., & Swanson, D. B. (2016). *Constructing written test questions for the basic and clinical sciences*. National Board of Medical Examiners. https://www.nbme.org/sites/default/files/2020-01/ConstructingWrittenTestQuestions_fifth_edition_2016.pdf [GS Search].
- Chico, V. J. S., Regino, A. G., Bonacin, R., & Reis, J. (2025). LLM-based automatic generation of multiple-choice questions with meaningful distractors. *Anais do XXXVI Simpósio Brasileiro de Informática na Educação*, 813–827. <https://doi.org/10.5753/sbie.2025.12675> [GS Search].
- Cho, Y., AlMamlook, R. E., & Gharaibeh, T. (2024). A systematic review of knowledge tracing and large language models in education: Opportunities, issues, and future research. *arXiv preprint arXiv:2412.09248*. <https://doi.org/10.48550/arXiv.2412.09248> [GS Search].
- Denzin, N. K. (1978). *The research act: A theoretical introduction to sociological methods* (2nd ed.). McGraw-Hill. [GS Search].
- Elkins, S., Kochmar, E., Cheung, J. C. K., & Serban, I. (2024). How teachers can use large language models and Bloom's taxonomy to create educational quizzes. <https://doi.org/10.48550/arXiv.2401.05914> [GS Search].

- Haladyna, T. M., Downing, S. M., & Rodriguez, M. C. (2002). A review of multiple-choice item-writing guidelines for classroom assessment. *Applied Measurement in Education*, 15(3), 309–333. https://doi.org/10.1207/S15324818AME1503_5 [GS Search].
- Hang, C. N., Tan, C. W., & Yu, P.-D. (2024). MCQGen: A large language model-driven MCQ generator for personalized learning. *IEEE Access*, 12, 102261–102273. <https://doi.org/10.1109/ACCESS.2024.3420709> [GS Search].
- Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., & Liu, T. (2025). A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Computing Surveys*. <https://doi.org/10.1145/3703155> [GS Search].
- Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., . . . Kasneci, G. (2023). Chat-GPT for good? on opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103, 102274. <https://doi.org/10.1016/j.lindif.2023.102274> [GS Search].
- Li, K., & Zhang, Y. (2024). Planning first, question second: An LLM-guided method for controllable question generation. *Findings of the Association for Computational Linguistics: ACL 2024*, 4715–4729. <https://doi.org/10.18653/v1/2024.findings-acl.280> [GS Search].
- Maity, S., Deroy, A., & Sarkar, S. (2023). Harnessing the power of prompt-based techniques for generating school-level questions using large language models. *Proceedings of the 15th Annual Meeting of the Forum for Information Retrieval Evaluation*, 30–39. <https://doi.org/10.1145/3632754.3632755> [GS Search].
- Mucciaccia, S. S., Paixão, T. M., Mutz, F. W., Badue, C. S., Souza, A. F., & Oliveira-Santos, T. (2025). Automatic multiple-choice question generation and evaluation systems based on LLM: A study case with university resolutions. *Proceedings of the 31st International Conference on Computational Linguistics*, 2246–2260. <https://aclanthology.org/2025.coling-main.154/> [GS Search].
- Quincozes, C. B., Molinos, D., Araújo, R. D., Quincozes, S., & Guedes, G. T. A. (2025). Engenharia de prompt para a geração automatizada de questões assistida por LLMs: Uma análise comparativa. *Anais do XXXVI Simpósio Brasileiro de Informática na Educação*, 1348–1361. <https://doi.org/10.5753/sbie.2025.12891> [GS Search].
- Scaria, N., Chenna, S. D., & Subramani, D. (2024a). Automated educational question generation at different Bloom’s skill levels using large language models: Strategies and evaluation. *Artificial Intelligence in Education*, 14830, 165–179. https://doi.org/10.1007/978-3-031-64299-9_12 [GS Search].
- Scaria, N., Chenna, S. D., & Subramani, D. (2024b). How good are modern LLMs in generating relevant and high-quality questions at different Bloom’s skill levels for Indian high school social science curriculum? *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, 1–10. <https://aclanthology.org/2024.bea-1.1/> [GS Search].
- Shu, C., Yao, N., Chen, Y., Wijeratne, V., Ma, L., Loo, J., Chai, K. K., Alam, A. S., & Abuelmaatti, A. (2025). AI-assisted multiple-choice questions generation with multimodal large language models in engineering higher education. *2025 IEEE Global Engineering Education*

- Conference (EDUCON)*, 1–9. <https://doi.org/10.1109/EDUCON62633.2025.11016449> [GS Search].
- Tran, A., Angelikas, K., Rama, E., Okechukwu, C., Smith, D. H., & MacNeil, S. (2023). Generating multiple choice questions for computing courses using large language models. *2023 IEEE Frontiers in Education Conference (FIE)*, 1–8. <https://doi.org/10.1109/FIE58773.2023.10342898> [GS Search].
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35, 24824–24837. [GS Search].
- White, J., Fu, Q., Hays, S., Sandborn, S., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., & Schmidt, D. C. (2023). A prompt pattern catalog to enhance prompt engineering with ChatGPT. *arXiv preprint arXiv:2302.11382*. <https://doi.org/10.48550/arXiv.2302.11382> [GS Search].
- Xu, H., Gan, W., Qi, Z., Wu, J., & Yu, P. S. (2024). Large language models for education: A survey. *arXiv preprint arXiv:2405.13001*. <https://doi.org/10.48550/arXiv.2405.13001> [GS Search].
- Yao, Z., Parashar, A., Zhou, H., Jang, W. S., Ouyang, F., Yang, Z., & Yu, H. (2025). MCQG-SRefine: Multiple choice question generation and evaluation with iterative self-critique, correction, and comparison feedback. *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 10728–10777. <https://doi.org/10.18653/v1/2025.naacl-long.538> [GS Search].