Robust LLM-as-a-Judge Validators for Assessing the Quality of Educational Exams

Renzo Degiovanni
renzo.degiovanni@list.lu
Luxembourg Institute of Science and Technology
Luxembourg

Sergio Morales
smoralesg@uoc.edu
Universitat Oberta de Catalunya
Barcelona, Spain

Miriam Coccia
miriam.coccia@list.lu
Luxembourg Institute of Science and Technology
Luxembourg

Robert Clarisó
rclariso@uoc.edu
Universitat Oberta de Catalunya
Barcelona, Spain

Jordi Cabot
jordi.cabot@list.lu
Luxembourg Institute of Science and Technology
Luxembourg

ABSTRACT

Large Language Models (LLMs) offer promising opportunities in the context of education, for example, to automate the exams' creation process. Although LLMs can save educators time and effort, their integration into educational products brings some risks that companies must mitigate.

In this paper, we discuss a set of quality criteria (e.g. correctness and clarity), grounded in the literature and validated by our industrial partner, relevant for designing educational exams. We also present a set of validators, using an LLM-as-a-judge approach, to automatically check the quality of educational exams. We evaluate their robustness on quality and (artificially generated) defective educational items, including 17 commercial and open-source LLMs.

Our experiments show that the most recent and larger models are very precise, trigger almost no false alarms, and are effective in detecting most quality issues. Moreover, open source Llama and DeepSeek models perform as well as GPT models, although Mistral seems unsuitable for this task. The promising results encourage us to focus on more general types of questions, for which we report relevant open challenges we aim to address in the short term.

ACM Reference Format:
Renzo Degiovanni, Sergio Morales, Miriam Coccia, Robert Clarisó, and Jordi Cabot. 2026. Robust LLM-as-a-Judge Validators for Assessing the Quality of Educational Exams. In *The 41st ACM/SIGAPP Symposium on Applied Computing (SAC '26), March 23–27, 2026, Thessaloniki, Greece.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3748522.3779926

1 INTRODUCTION

In the field of education, tests are widely used to gather insights about students' knowledge, skills, and abilities in relation to defined learning objectives. These tests can include questions (items) in diverse formats: multiple-choice, true or false, short answer or essays. Regardless of their format, assessment items should provide a relevant and accurate measure of the level of knowledge and proficiency of the learner. As a result, item quality is essential to ensure the validity of test.

The quality of an assessment item or test can be studied from different points of view. Standardized educational assessment tests consider quality criteria such as correctness (there are no errors in the problem formulation), completeness (the problem formulation includes the relevant data and context to answer the question), clarity and lack of ambiguity. However, many other quality properties may be considered, such as the desired level difficulty, accessibility for students with special needs, or the absence of biases.

Creating high quality assessment tests is a time-consuming task that requires experience and domain expertise. As a result, educational institutions and professionals are continuously looking into ways to facilitate the creation of such tests. A promising approach in this direction is the use of Large Language Models (LLMs) [10], which are able to generate human-like textual content according to the instructions received via an input prompt. However, LLMs have several weaknesses: they are prone to hallucinations [18] and biases [12], and may sometimes fail to follow the input instructions. As a result, LLM-generated tests should be reviewed for quality purposes before being deployed in any real-world educational setting [3]. Fortunately, LLMs can also be used as a powerful automated validation tool, for example, using the paradigm called *LLM-as-a-judge* [35], where the LLM critically reviews a piece of content using some user-defined criteria.

In this paper, we consider the automatic validation of educational tests using LLMs. To this end, we define a comprehensive set of quality criteria for educational tests that should be reviewed in order to ensure their suitability. We then define a set of automatic validators,using *LLM-as-a-judge* strategies, for checking those quality properties and empirically evaluate their effectiveness. This is done in collaboration with the company Open Assessment Technologies S.A. (OAT[1]) that provides advanced assessment solutions for education in 194 countries and 30+ languages, which helped us to review and validate the proposed quality criteria.

SAC'26, March 23-27, 2026, Thessaloniki, Greece

ACM ISBN 979-8-4007-2294-3/2026/03.
https://doi.org/10.1145/3748522.3779926

[1] https://www.taotesting.com/

We empirically evaluate our validators on 17 commercial and open-source LLMs from Open AI, Meta Llama, Mistral AI, and DeepSeek. Our experiments are performed on the ScienceQA [22] dataset which contains multiple-choice questions collected from elementary and high school science curricula. Since this dataset was manually analyzed and validated, we assume that the questions are of good quality and thus, we expect that all the validators to pass. However, to evaluate our validators' robustness, we design property-specific metamorphic transformations [30] (a.k.a. mutations [26]) to automatically generate defective questions for which we expect the validators to fail on specific quality properties. Hence, we rely on standard metrics, such as precision and recall, to objectively measure the robustness of our validators. .

The empirical results show that more recent models (e.g. gpt-4o, llama3.3:70b-instruct, and deepseek-r1:32b-qwen-distill) are very precise, triggering almost no false alarms. This means that, in practice, when the validators using some of these models fail on a particular question, it is almost certain that there is an issue affecting some of the quality criteria. This is considered a key advantage for our industrial partner, as it will avoid educators wasting time and effort in analyzing good questions. We also observe that our LLM-based validators are very effective at detecting most of the issues (~100% of recall), affecting more objective properties such as the Correctness and Difficulty. However, our validators can miss many of the issues affecting more subjective properties, such as the Scope and Background of the questions that may depend on contextual information not present in the prompts (e.g. a detailed curricula), obtaining an overall recall of ~60%.

In summary, our experiments show that open source models from Llama and DeepSeek can perform as well as the commercial models from Open AI. This can encourage companies to integrate open source LLMs into their educational products, mainly in those cases in which the exams cannot be shared. We also observe that, although Mistral models also produce a low number of false positives, they show a poor performance in identifying the injected quality issues, making them less suitable for our purposes. Finally, the property-specific metamorphic transformations we use to assess the robustness of our validators, helped us a lot to refine the initial versions of our prompts, significantly improving their performance.

Although our experiments focus on multiple-choice questions, the proposed quality indicators and validators are applicable to other types of assessment items that we plan to study in the short term. In collaboration with our industrial partner, we also plan to adapt them to multimodal contexts to analyze quality properties of questions involving images or audio, which will demand the integration of multimodal LLMs, such as Gemini and GPT-4V. We also plan to explore more sophisticated prompt engineering techniques, such as Chain-of-Thought Prompting [28] and Retrieval-Augmented Generation [21], to potentially improve their performance.

2 BACKGROUND AND RELATED WORK

In the literature, the problem of synthesizing assessment items is described using different terms, such as *automatic question generation* or *automatic item generation* [5, 8, 9, 20, 32].

The generation process may focus on a given discipline, domain, or learning outcome, and it can be based on a set of input knowledge sources. Different types of assessment items can be generated [9], among which multiple-choice questions (MCQ) are the most frequently explored.

MCQs are characterized by at least three elements: the *stem* (the problem statement), the correct *answer*, and the *distractors* (incorrect answers proposed as alternatives). It is also possible to provide additional information about the question, such as learning outcomes, an academic level, or a potential solution, among others.

Originally, item generation relied on a combination of Natural-Language Processing (NLP) and information retrieval techniques to analyze an input text, followed by a wide variety of approaches (e.g., pattern matching, knowledge representation and machine learning) to propose suitable question stems, answers, and distractors [5, 9, 20]. The introduction of LLMs has sparked significant interest in their application to the item generation problem [6, 23, 29, 33].

A major challenge in automatic item generation is ensuring the quality of the generated items [11, 14, 27]. Thus, many approaches include an explicit evaluation stage at the end of the generation process to filter low-quality questions. This validation can either be performed manually by human experts [1, 19, 27], or it may rely on automatic procedures [2, 19], *e.g.*, filtering questions that are too short or using ontologies to predict the difficulty of a question.

In this direction, *LLM-as-a-judge* is an increasingly common strategy that leverages an LLM to evaluate the outputs of another model. This approach enables the semi-automation of qualitative analysis, thereby improving the scalability and cost-efficiency of evaluation workflows [35]. An LLM-as-a-judge can be used for either scoring or rating a single output, comparing two or more outputs and choosing which is better, or giving structured, qualitative feedback [7]. Nevertheless, this strategy is reported to face challenges such as biases, inconsistency in responses to the same output, and overfitting to certain prompt designs or domains [16, 34]. This approach can also be applied to the evaluation of automatically generated items [19, 33], but it is typically applied to a specific quality criteria and using a particular LLM. To provide a holistic evaluation of quality, Section 3 defines a set of quality criteria and Section 4 details how we have relied on the reasoning capabilities of an LLM-as-a-judge to assess them in the generated tests, comparing the results achieved by several LLMs.

3 STANDARDIZED EDUCATIONAL EXAMS QUALITY PROPERTIES

As introduced in Section 2, the quality of the generated questions is a key concern of automated item generation. Quality may refer to different dimensions [15], from the adequacy of the question stem with respect to the input text, discipline, learning outcome or intended academic level; the clarity and readability of the question; the correctness of the answer (and the lack thereof for distractors); or the suitability of distractors.

However, there is no universally accepted set of standard quality criteria for generated assessment items. For instance, [27] uses human experts to manually evaluate five quality criteria: (1) the relationship between the stem and the key, (2) the clarity of the stem, (3) the lack of cues that makes the answer obvious, (4) an homogeneous length among alternative answers and (5) the plausibility of distractors. In contrast, [14] used eight criteria, which

include the relevance of the assessed topic, and the alignment of the question's topic with the curricula. Other works such as [24, 31] also consider distractors, checking that they do not overlap with the answer but at least one of them is sufficiently close.

Research in education has also studied quality criteria for MCQs. In this sense, [17] presents a taxonomy of 31 guidelines for writing MCQs, which focus on five areas: the choice of content; formatting; style; the definition of the stem (clarity, conciseness, avoiding negatives); and writing the alternatives (plausibility, lack of clues, correctness, ...). Some measures such as the effectiveness of distractors can be measured *a posteriori* [13], discarding distractors that were not selected by students taking the test. In this way, this measure can be useful to revise and improve a test, but it is not applicable for newly created ones.

Recent works emphasize on the relevance of analyzing item quality, and propose taking advantage of the flexibility of LLMs to improve quality evaluations by checking quality properties that focus higher-level issues such as learning outcomes [33]. For instance, it is possible for LLMs to assess properties like the *authenticity* of an item, *i.e.*, whether it reflects a real-world situation instead a theoretical or academic scenario; or its *accessibility* to students with special needs such as impaired sight. As a result, based on quality criteria in the literature and the capabilities of LLMs, we have proposed a list of quality criteria for assessment items. This list has also been validated by our industrial partner as suitable for their needs in real-world scenarios.

Table 1 presents our proposed list of quality criteria for evaluating educational tests. Some of the proposed criteria require additional information to be evaluated, as achieving the quality criteria depends on the intended target. For instance, in order to decide if students have sufficient background to answer a particular question, it is necessary to specify the target educational level and knowledge available to them: within the same discipline, a question may be suitable for a master's student but inadequate for a high school student. Some quality criteria will be omitted from our analysis, as they either focus on questions with open answers (unlike MCQs) or complete tests.

4 LLMS AS QUALITATIVE VALIDATORS

This section presents our LLM-as-a-judge approach to automate the quality check of educational questions. Then, we describe the experimental setup and the ScienceQA dataset, a well-established resource for evaluating educational Q&A tasks, as the foundation for our analysis. Finally, we compare the effectiveness of different LLMs in performing this evaluative task, assessing their ability to reliably judge the quality of educational questions. Notice that, although our industrial partner envisions applying these validators to analyze LLM-generated questions, they are general enough to be applied to manually written questions as well.

4.1 Prompt

Listing 1 presents the prompt of our LLMs-as-judges validators. It takes three primary components as input, namely: `QUESTION`, `CHARACTERISTICS` and `QUALITY_CRITERIA`. Either manually or automatically generated, we provide the educational question to be evaluated in raw text format, within the placeholder `{question}`.

Listing 1: Prompt used for the LLM-as-a-judge validator.

```
Given the CHARACTERISTICS below and the QUESTION given
by an AI assistant as a response to the CHARACTERISTICS
Does the QUESTION comply with the following
QUALITY_CRITERIA (which are provided in JSON format,
with keys \"property\" and \"definition\")?
Notice that this QUESTION will be used for assessing
Student Grade Level: {grade}.

CHARACTERISTICS: ```{characteristics}```
QUESTION: ```{question}```
QUALITY_CRITERIA: ```{criteria}```
```

In `{characteristics}`, we include metadata such as the target students' grade level, demographical information, the subject and topic, and any other information for enriching the context provided to the LLM. Finally, in the *{criteria}* placeholder, we include the definition of the quality criteria in JSON format. For each criterion, we provide an identifier *property* and a *definition* that describes its semantics. Table 1 summarizes the quality criteria we analyze, previously revised and validated by our industrial partner. Listing 2 shows the structure of the different inputs for one of the examples used in our experiments.

Finally, the response generated by the LLM-as-a-judge must adhere to a structured, machine-readable format, for enabling an automated post-processing integration of results with analytic tools and human-in-the-loop reviews. In order to do so, the prompt includes further instructions on the required output format that are automatically generated by Pydantic parser[2]. The system expects a list of *(property, valid, reasoning)* tuples, where: *property* should reference an input qualitative criterion; *valid* should be True if the question satisfies the property, and False, otherwise; and *reasoning* should contain an explanation of why the question does not satisfy the property, and should be empty if the property is satisfied.

4.2 Empirical Evaluation Setup

Dataset. We start our empirical evaluation by studying the performance of our validators on the ScienceQA [22] dataset, widely used for assessing educational Q&A tasks. ScienceQA contains 21,208 multimodal questions on subjects related to natural science, language science, and social science. We focus our evaluation on multiple-choice questions from the testing set that only contains textual information, that is, questions including images are excluded. After this filtering, our final dataset includes 2,071 questions.

For each data point, ScienceQA provides the following information: the *question* with the *options* and the corresponding correct *answer*; the *lecture* that provides context to solve the question; it can also include some *hint* and a justification of the *solution*; the target student *grade* (level); and the *subject*, *topic*, and *skill* targeted by the question. Listing 2 shows a multiple-choice question (with id 22) taken from the dataset.

Metrics. Notice that the questions in the ScienceQA dataset were collected from elementary and high school science curricula, annotated with grounded lectures and detailed explanations that provide enough context for arriving at the correct answer. Since this dataset has been manually analyzed and validated, we assume

[2]https://python.langchain.com/docs/how_to/output_parser_structured

Table 1: Proposed quality criteria to evaluate automatically generated exams.

Property	Input	Definition
Scope	Educational level, learning outcome	The activity is suitable to assess a target learning outcome at a given Student Grade Level.
Background	Educational level, prior knowledge	Students at the given Grade Level have the required background to understand and solve the activity.
Clarity		The activity describes the context, task and intended outcome unambiguously. The solution is clear and all the steps, decisions and alternatives are outlined.
Conciseness		The problem statement is succinct, avoiding repetition and verbosity.
Reliability[a]		It is possible to consistently evaluate the correctness of a candidate solution.
Discrimination		Distractors are effective at drawing incorrect answers.
Correctness		The problem statement does not contain errors, missing or inconsistent data. The solution fulfills all the requirements and contains no errors or omissions. All valid solutions have been characterized & distractors are indeed invalid.
Difficulty	Intended difficulty	The activity is neither obvious nor impossible for a given Student Grade Level: it is challenging yet feasible.
Workload[b]	Intended workload	The activity takes an average student, for a given Student Grade Level, a reasonable amount of time to be solved.
Format	Intended format	The description of the activity adheres to the requested format or template.
Accessibility	Special needs	The activity can be understood and solved by students with special needs.
Authenticity	Learning outcome	The activity captures a realistic scenario that is relevant to students and the field.
Inclusivity		The activity does not contain inappropriate or biased content.
Validity[b]	Learning outcome	The activity tests a relevant skill considering all its relevant perspectives.

[a] Relevant only for assessment items with open answers (unlike MCQs). [b] Relevant only for complete tests (not individual items).

Listing 2: QUESTION and CHARACTERISTICS example.

```
CHARACTERISTICS: ```Student Grade Level: 8
Subject: language science
Topic: reference-skills
Category: Reference skills
Skill: Use guide words```

QUESTION: ```Question: Which word would you find on a
dictionary page with the following guide words?
shot - suit
Options: (A) service (B) stockade
Answer: The answer is B.
Lecture: Guide words appear on each page of a
dictionary. They tell you the first word and last word
on the page. The other words on the page come between
the guide words in alphabetical order. To put words in
alphabetical order, put them in order by their first
letters. If the first letters are the same, look at the
second letters. If the second letters are the same,
look at the third letters, and so on. If one word is
shorter, and there are no more letters to compare, then
the shorter word comes first in alphabetical order. For
example, be comes before bed.
Solution: Put the words in alphabetical order. Since
stockade is between the guide words shot - suit, it
would be found on that page.```

QUALITY_CRITERIA: ```{'Scope': 'The activity is
suitable to assess a target learning outcome at a given
Student Grade Level.', 'Background': ...' ... }```
```

that the questions are of good quality, therefore, our expectation is that all validators should pass when applied to these questions. Hence, to assess our validators performance, we first measure the percentage of questions for which the LLM passes each specific quality property. This will help us understand whether some LLMs are more suitable than others to analyze specific properties.

Table 2: LLMs studied.

OpenAI
gpt-4o, gpt-4o-mini, gpt-4-turbo, gpt-3.5-turbo-1106
Meta Llama
llama3.3:70b-instruct, llama3.1:70b-instruct, llama3.1:8b-instruct, llama3:70b-instruct, llama3:8b-instruct
Mistral AI
mistral:7b-instruct-v0.3, mixtral:8x22b-instruct, mistral:7b-instruct-v0.2, mixtral:8x7b-instruct
DeepSeek
deepseek-r1:32b-qwen-distill, deepseek-r1:7b-qwen-distill, deepseek-r1:70b-llama-distill, deepseek-r1:8b-llama-distill

We also report the general *false positive rate* of each LLM measured as the percentage of property checks that fail on the entire ScienceQA dataset (since we expect them to pass). For our industrial partner, it is very important to keep this number low, since each false positive will require a manual inspection by a human expert to determine if there is an issue in the question, and fix it accordingly.

Selected LLMs. To ensure that our conclusions are not caused by the selection of a particular LLM, our experiments compare the outcome of our LLM-as-a-judge prompt using 17 LLMs, including 4 models from Open AI, 5 from Meta Llama, 4 from Mistral AI, and 4 from DeepSeek (see Table 2). In this way, we are also able to identify which LLM performs better for this task.

Data Availability. All the scripts and results are available online: https://github.com/rdegiovanni/LLMValidatorsTesting.

4.3 LLMs' Effectiveness

Table 3 summarizes, for each LLM, the percentage of questions from the ScienceQA dataset that pass the specific quality property. The last column reports the false positive rate (FPR) of each LLM, *i.e.*, the percentage of failed checks among all the questions.

First, we can observe that, for every quality property, there is at least one validator (LLM) for which all the questions passed. We also notice that, in general, larger models are more effective (lower false positive rate) than smaller ones. Reliability, Inclusivity, Scope, Background, and Correctness show low variance across validators, which indicates that these criteria are consistently well-understood by models. On the other hand, Conciseness, Authenticity, Validity and Discrimination exhibit high variability in accuracy, suggesting that they require more nuanced and deeper understanding, which lower-tier versions of Llama and DeepSeek models often miss.

Results show that Open AI models have a similar good performance overall, with gpt-3.5-turbo-1106 being the worst performing. Notably, all GPT models are less robust with respect to the Conciseness property, producing the most false positives, which indicates that they might not be good candidates to perform this specific quality check. In the case of Llama models, larger models (70B) clearly show much better performance than smaller variants (8B). In particular, more recent releases (3.3. and 3.1) lead to a false positive rate of 1%, making them good candidates for adoption in practice.

The best performing model is deepseek-r1:32b-qwen-distill, where almost all questions pass every quality check, leading to almost 0% FPR, although the other variants of DeepSeek performed poorly. On the other hand, Mistral AI models have a comparable performance with the best performing models, with the only exception of mistral:7b-instruct-v0.3 with a FPR of 16%.

Two key observations emerged from these results. On the one hand, the best performing models raised very few false alarms, *i.e.*, very few questions failed specific properties, which is a desired behavior in practice for our industrial partner. Having validators that frequently fail on quality questions would cause the users to distrust their judgment, and make them uninteresting. On the other hand, *open-source* models obtained very good and promising performance, comparable with the GPT proprietary models. This evidence can help increase and ease their adoption in education.

5 ASSESSING VALIDATORS' ROBUSTNESS

We just observed that the best performing LLMs are reliable on quality questions, triggering very few false alarms. Now we study the robustness of our validators in detecting quality issues of defective educational questions. To do so, we rely on the metamorphic testing methodology [30] and design specific *metamorphic transformations* that modify the ScienceQA dataset questions in such a way that a quality issue is introduced with the aim of breaking specific properties. Then, we use these (artificially generated) defective educational questions (a.k.a. mutations [26]) to assess the robustness of our validators (LLMs), measured in terms of the standard performance metrics precision and recall.

5.1 Property-specific defect injection

Correct Answer Transformation. This simple transformation consists of (incorrectly) selecting one of the distractors in a multiple-choice question as the correct answer. This defective question is expected to break the Correctness property, and it is used as the *metamorphic oracle* to determine if our validators are able to detect the injected quality issue. Following with the multiple-choice question given in Listing 2, this metamorphic transformation would select option `(A) service` as the correct answer to the question, which is completely incorrect since the word 'service' cannot be found in the dictionary between the words 'shot' and 'suit'.

Students Grade Transformation. This transformation takes questions for primary school students (between grades 1 and 6) and transforms them into questions for students in the last year of secondary school (grade 12). Vice versa, it takes questions for secondary school (students between grades 7 and 12) and transforms them into questions for students in the first year of primary school (grade 1). We expect this transformation to create defective questions that break four properties: Scope, Background, Difficulty and Workload, which is used as the expected *metamorphic oracle*. Following with Listing 2, the original question was proposed for students of grade 8 of secondary school, this transformation will change it to students of the first grade of primary school (grade 1). A student in first grade is typically learning to read and write, so clearly he/she does not have enough scope and background to understand how to use the dictionary, making the transformed question too difficult and almost impossible for them to answer.

5.2 Evaluation Metrics

As our validators will determine whether a given educational question satisfies or not a particular quality property of interest, they may produce four possible outputs: given a correctly formulated question (taken from the ScienceQA dataset), if the validator identifies no quality issue, then it is a true negative (TN); otherwise, it is a false positive (FP). On the other hand, given a defective question that violates a particular quality property of interest (*e.g.*, generated with our metamorphic transformations), if the validator identifies the issue, then it is a true positive (TP); otherwise, it is a false negative (FN). From these, we can compute the traditional evaluation metrics such as *Precision*, *Recall*, and the *F-1 score* to evaluate the prediction performance of our validators.

$$Precision = \frac{TP}{TP + FP} \qquad Recall = \frac{TP}{TP + FN}$$

$$F1\text{-}score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Precision indicates the ratio of correctly predicted positives over all the considered positives. *Recall* indicates the ratio of correctly predicted positives over all the actual positives. *F-1 score* indicates the weighted harmonic mean of Precision and Recall.

5.3 Robustness Results

Table 4 reports, for each of the quality properties affected by the metamorphic transformations, the prediction performance of our validators. Notice that we highlight the best performing model (highest F1) in bold, and the second one with an underscore.

Correctness. This is highly relevant for our industrial partner. Luckily, we observe that most of the LLMs, mainly the large ones, have a high precision in detecting issues affecting the Correctness

Table 3: Per-trait LLMs' accuracy on the ScienceQA dataset.

Model	Scope	Background	Clarity	Conciseness	Reliability	Discrimination	Correctness	Difficulty	Workload	Format	Accessibility	Authenticity	Inclusivity	Validity	FPR
gpt-4o	1.00	1.00	0.88	0.67	1.00	0.91	0.94	1.00	1.00	0.94	0.93	0.96	1.00	0.98	0.06
gpt-4o-mini	1.00	1.00	0.91	0.52	1.00	1.00	0.96	0.98	1.00	0.99	1.00	1.00	1.00	0.98	0.05
gpt-4-turbo	1.00	1.00	0.98	0.76	1.00	0.95	0.98	0.99	0.98	0.94	0.94	0.99	1.00	0.99	0.04
gpt-3.5-turbo-1106	0.84	0.98	0.93	0.83	0.93	0.90	0.83	0.90	0.93	0.91	0.97	0.92	0.98	0.88	0.09
llama3.3:70b-instruct	1.00	1.00	0.99	0.98	1.00	0.99	1.00	1.00	1.00	1.00	0.96	1.00	1.00	1.00	**0.01**
llama3.1:70b-instruct	1.00	0.99	0.99	0.99	1.00	0.98	0.99	0.99	1.00	1.00	0.99	0.99	1.00	1.00	**0.01**
llama3:70b-instruct	0.99	1.00	1.00	0.95	1.00	0.69	1.00	0.99	1.00	0.99	0.80	0.90	1.00	0.99	0.05
llama3.1:8b-instruct	0.32	0.59	0.5	0.46	0.60	0.55	0.52	0.55	0.50	0.70	0.56	0.37	0.82	0.13	0.49
llama3:8b-instruct	0.10	0.80	0.63	0.56	0.71	0.54	0.69	0.38	0.61	0.81	0.87	0.20	0.93	0.17	0.43
deepseek-r1:32b-qwen-distill	1.00	1.00	1.00	0.99	1.00	1.00	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	**0.00**
deepseek-r1:7b-qwen-distill	0.61	0.42	0.30	0.50	0.43	0.34	0.32	0.45	0.54	0.48	0.44	0.36	0.50	0.28	0.57
deepseek-r1:70b-llama-distill	1.00	0.82	0.57	0.71	0.92	0.41	0.88	0.76	0.90	0.66	0.56	0.73	1.00	0.77	0.24
deepseek-r1:8b-llama-distill	0.70	0.73	0.74	0.74	0.76	0.74	0.67	0.73	0.68	0.76	0.82	0.61	0.82	0.62	0.28
mixtral:8x22b-instruct	1.00	1.00	1.00	0.74	1.00	0.90	0.99	0.93	0.86	0.83	0.96	0.84	1.00	0.94	0.07
mixtral:8x7b-instruct	0.99	1.00	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	**0.01**
mistral:7b-instruct-v0.3	0.72	0.73	0.95	0.95	0.94	0.94	0.93	0.72	0.85	0.85	0.92	0.70	0.93	0.67	0.16
mistral:7b-instruct-v0.2	0.87	0.86	1.00	1.00	1.00	1.00	1.00	1.00	0.97	1.00	1.00	0.99	1.00	0.86	0.03

property. This is a very promising result, suggesting that these LLMs will report few false alarms to educators, not requiring much of their manual intervention. However, when we look at the recall, we can observe that only a few models can detect most of the quality issues. In particular, llama3.3:70b-instruct can detect 92% of the incorrect transformations, while gpt-4o and deepseek-r1:32b-qwen-distill only 85% and 80%, respectively. The rest of the LLMs have a high false negative rate (leading to lower recall), missing many of the issues affecting the correctness of the questions.

Scope, Background, Difficulty and Workload. We observe that most GPT models (except for gpt-3.5-turbo-1106), 70B Llama models, and deepseek-r1:32b-qwen-distill obtain very high precision (∼100%), producing almost no false positives, a desired behavior for our industrial partner. However, when we look at the recall, LLMs trigger significantly more false negatives, mainly for Workload and Background properties. For Workload, llama3.3:70b-instruct performs best with an F1-score of 52%, but a recall of 36%, meaning that it misses the issue in 64% of the cases. For Background, gpt-4o-mini obtains an F1-score of 59% with a recall of 41%, missing 59% of the issues affecting this property. In the case of Scope, LLMs show a relatively better performance, where gpt-4o and llama3.3:70b-instruct models obtain a promising F1-score of 80% and 69%, respectively, missing 32% and 41% of the quality issues.

Finally, LLMs are very effective in analysing Difficulty, where the best performing Llama models only miss between 2-3% of the issues, leading to an F1-score on ∼98%. gpt-4o also has a good performance on this property, although the recall is lower (88%), missing 12% of the issues. This may suggest that LLMs can perform better when the task under analysis is more objective (e.g. Correctness) or there is clear competing/contradicting information in the question (e.g., between the age of the students and the Difficulty of the question), but their performance reduces when dealing with more subjective tasks, such as Background and Scope, which may require more contextual information.

Average. The most robust models are gpt-4o and llama3.3:70b-instruct, obtaining a 75% of F1-score, on average. Other GPT models (gpt-4o-mini and gpt-4-turbo) as well as llama3.1:70b-instruct also show promising performance with a ∼60% of F1-score, on average. deepseek-r1:32b-qwen-distill is also a promising option with a F1 of 55%. In the case of Mistral AI models, we did not observe a good performance comparable with the other models, which may suggest that are not adequate for this task.

6 LESSONS LEARNED AND OPEN CHALLENGES

Open Source vs Commercial LLMs. Our results show that open source models, in particular Llama and DeepSeek, can perform as well as the commercial models from Open AI. Considering that some companies may be reluctant to share their educational tests, because these may contain students' sensitive information or are expensive to generate (typically by domain experts), locally deployed open source models sound like a good alternative to mitigate this issue.

False Positives and False Negatives. The best performing models produced a marginal number of false positives. This means that, in practice, whenever a validator fails, it is almost certain that there is an issue on the educational question. This will help educators save time and effort, not waste time analyzing good questions, which is a key advantage envisioned by our industrial partner.

Regarding false negatives, our validators are more effective at detecting issues affecting more objective properties such as Correctness rather than more subjective properties such as Background. In the case of Correctness, the prompt already provides all the relevant information (the theoretical concepts and context) to the LLM to determine which of the options in the multi-choice should be selected. Hence, our validators can detect most of the issues in the detective

Table 4: LLMs' Robustness on Correctness, Scope, Background, Difficulty and Workload quality properties.

	Correctness			Scope			Background			Difficulty			Workload			Average		
Model	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
gpt-4o	0.93	0.85	0.89	0.99	0.68	**0.80**	1.00	0.4	0.57	0.99	0.88	0.93	1.00	0.25	0.40	0.98	0.61	**0.75**
gpt-4o-mini	0.94	0.53	0.68	1.00	0.41	0.59	1.00	0.41	**0.59**	0.98	0.53	0.68	1.00	0.32	0.49	0.98	0.44	0.61
gpt-4-turbo	0.97	0.75	0.85	1.00	0.21	0.35	1.00	0.29	0.45	0.98	0.65	0.78	1.00	0.31	0.47	0.99	0.44	0.61
gpt-3.5-turbo-1106	0.69	0.37	0.48	0.00	0.00	0.00	0.00	0.00	0.00	0.66	0.20	0.30	0.50	0.05	0.09	0.37	0.12	0.19
llama3.3:70b-instruct	1.00	0.92	**0.96**	1.00	0.53	0.69	1.00	0.22	0.37	1.00	0.95	0.97	1.00	0.36	**0.52**	1.00	0.60	**0.75**
llama3.1:70b-instruct	1.00	0.65	0.79	1.00	0.31	0.47	1.00	0.12	0.22	1.00	0.97	**0.98**	1.00	0.08	0.15	1.00	0.43	0.60
llama3:70b-instruct	0.98	0.07	0.13	1.00	0.18	0.31	1.00	0.02	0.04	1.00	0.87	0.93	1.00	0.01	0.02	1.00	0.23	0.37
llama3.1:8b-instruct	0.57	0.61	0.59	0.45	0.39	0.42	0.71	0.49	0.58	0.67	0.84	0.75	0.31	0.20	0.24	0.54	0.51	0.52
llama3:8b-instruct	0.60	0.38	0.47	0.30	0.33	0.31	0.45	0.09	0.15	0.57	0.92	0.70	0.10	0.04	0.05	0.40	0.35	0.38
deepseek-r1:32b-qwen-distill	0.99	0.80	0.88	1.00	0.04	0.08	1.00	0.27	0.42	1.00	0.66	0.80	1.00	0.11	0.20	1.00	0.38	0.55
deepseek-r1:7b-qwen-distill	0.47	0.48	0.47	0.26	0.09	0.14	0.39	0.40	0.39	0.42	0.22	0.29	0.22	0.05	0.08	0.35	0.25	0.29
deepseek-r1:70b-llama-distill	0.77	0.39	0.52	1.00	0.04	0.08	0.48	0.21	0.29	0.76	0.79	0.78	0.57	0.08	0.14	0.72	0.30	0.42
deepseek-r1:8b-llama-distill	0.53	0.32	0.40	0.26	0.08	0.12	0.00	0.00	0.00	0.54	0.16	0.25	0.25	0.02	0.04	0.32	0.12	0.17
mixtral:8x22b-instruct	0.97	0.40	0.57	0.00	0.00	0.00	1.00	0.04	0.08	0.88	0.23	0.37	0.7	0.17	0.27	0.71	0.17	0.27
mixtral:8x7b-instruct-v0.1	0.92	0.17	0.28	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.02	0.03	0.00	0.00	0.00	0.38	0.04	0.07
mistral:7b-instruct-v0.3	0.83	0.13	0.23	0.33	0.04	0.07	0.00	0.00	0.00	0.40	0.05	0.08	0.33	0.02	0.04	0.38	0.05	0.09
mistral:7b-instruct-v0.2	1.00	0.01	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.20	0.00	0.00

questions in which the selected answer is one of the distractors. Something similar occurs with Difficulty, in which the LLM is sensitive enough to determine that questions for the secondary school are not adequate for grade 1, and vice versa, questions for primary are not adequate for grade 12. Although promising, these are edge cases, and we plan to study our validators' performance on more subtle metamorphic transformations (e.g., by increasing/decreasing the target grade by one or two years).

In the case of the Scope, Background, and Workload properties, we observe that the recall is not that high, for which many issues affecting these properties were not detected. As discussed with our industrial partner, we believe that we can improve the performance of our validators in this respect by providing better context regarding the students' metadata (e.g., country, education system, etc.) and curriculum according to a given level. While the ScienceQA dataset does not provide more information than the students grade, we are planning to reproduce same experimentation on calibrated data our industrial partner will provide us in the near future. Another possibility we are considering is to include information regarding the expected curriculum for particular levels, such as the standard TIMSS [4], that can improve LLMs' predictions.

Injected Issues Improved the Prompts. In our initial experimentation, we evaluated our validators only on the ScienceQA dataset, where very few false positives were triggered. This encouraged us to artificially introduce quality issues in the questions, by applying metamorphic transformations, to assess the robustness of our validators. Surprisingly, we observed that our initial versions of the prompts were not effective at all to detect the issues related to Scope, Background and Workload.The main strategy that help us to improve the performance of our prompts was to use exactly the same keywords, in particular "Student Grade Level", in both the properties definition and the given students characteristics. This helped the LLMs to better semantically link the quality properties and the students' level, significantly reducing the number of false negatives. In the future, we plan to design more property-specific metamorphic transformations to test and improve their robustness.

Advanced Prompting. We also seek to explore sophisticate prompt engineering techniques to potentially improve our validators' performance. We plan to study Few-Shot and Chain-of-Thought Prompting [28], and Retrieval-Augmented Generation [21], that have been shown to be effective in improving related LLM-based tasks.

LLMs Complementarity. We have observed that LLMs detect different issues in the same questions. Hence, we wonder whether we can combine multiple LLMs in such a way that we can reduce mainly the false negative rate, i.e., if we can detect most of the quality issues injected. Of course, adding more LLMs comes at a cost, so we plan to study what are the most cost-effective combinations.

Generalization to Other Types of Questions. Our quality criteria are general enough that also apply to other kinds of test assessment items different from multiple-choice questions, such as fill-in-the-blanks, true-false questions, etc. We plan to extend our experimental evaluation in this regard in the short term.

Adaptation to Multimodal Contexts. Educational tests can also include images, audio or video, to provide additional sources of information to the students. In collaboration with our industrial partner, we are currently defining relevant quality criteria for multimodal educational questions. While some quality properties are specifically related to the image/audio/video source, others focus on the consistency between the image/audio/video and the textual part. Our validators in this case will need to integrate multimodal LLMs, such as Gemini and GPT-4V, to analyze these multimodal quality criteria. To test and improve them, we plan to design specific metamorphic transformations to the multimodal setting, by mutating not only the text, but also the images/audio/video.

Discrimination in K-12 Contexts. We acknowledge the limitations of LLMs in K–12 contexts, including hallucinations, opaque decision-making, and embedded social biases. These risks are particularly critical since education is a high-risk domain as per the EU AI Act. We underscore the importance of providing means to assess and mitigate fairness issues [25] in LLM-based tools for education.

7 CONCLUSION

This work discussed the relevant quality properties of educational items, grounded in the literature and validated by our industrial partner, and proposed LLM-based validators to automate their checking. We empirically evaluated their robustness and integrated 17 state-of-the-art LLMs from four different families, including commercial and open-source models. We observed that, most recent and larger versions of Llama, DeepSeek and GPT models triggered almost no false positives, and identified most of the (artificially injected) quality issues. Overall, models that yielded best results across quality criteria are the most reliable and, notably, *open-source* models demonstrated competitive performance, which indicates their suitability for adoption in education. Moreover, we observed that their performance degrades on their smaller/older variants, and Mistral AI models showed a poor performance, not suitable for this task.

ACKNOWLEDGMENTS

This work has been partially funded by the RDI Law project "Innovations for 21st Century Assessment Authoring," financed by the Luxembourg Ministry of the Economy; the Luxembourg National Research Fund (FNR) PEARL program (grant agreement 16544475); and the project PID2023-147592OB-I00 "SE4GenAI," funded by MCIN/AEI/10.13039/501100011033.

REFERENCES

[1] Itziar Aldabe, Maddalen Lopez de Lacalle, Montse Maritxalar, Edurne Martinez, and Larraitz Uria. 2006. ArikIturri: an automatic question generator based on corpora and NLP techniques. In *Intelligent Tutoring Systems*. Springer Berlin Heidelberg, 584–594.

[2] Samah AlKhuzaey, Floriana Grasso, Terry R. Payne, and Valentina Tamma. 2024. Towards automatic evaluation of questions generated from ontologies. In *1st Workshop on Automated Evaluation of Learning and Assessment Content* (CEUR Workshop Proceedings). Vol. 3772. CEUR-WS.org.

[3] David Baidoo-Anu and Leticia Owusu Ansah. 2023. Education in the era of generative artificial intelligence (AI): understanding the potential benefits of ChatGPT in promoting teaching and learning. *SSRN Electronic Journal*.

[4] Boston College, TIMSS & PIRLS International Study Center. 2024. TIMSS 2023 Encyclopedia: Education Policy and Curriculum in Mathematics and Science. Tech. rep.

[5] Dhawaleswar Rao CH and Sujan Kumar Saha. 2020. Automatic multiple choice question generation from text: A survey. *IEEE Transactions on Learning Technologies*, 13, 1, 14–25.

[6] Kuang Wen Chan, Farhan Ali, Joonhyeong Park, Kah Shen Brandon Sham, Erdalyn Yeh Thong Tan, Francis Woon Chien Chong, Kun Qian, and Guan Kheng Sze. 2025. Automatic item generation in various STEM subjects using large language model prompting. *Computers and Education: Artificial Intelligence*, 8, 100344.

[7] Nadezhda Chirkova, Tunde Oluwaseyi Ajayi, Seth Aycock, Zain Muhammad Mujahid, Vladana Perlić, Ekaterina Borisova, and Markarit Vartampetian. 2025. LLM-as-a-qualitative-judge: automating error analysis in natural language generation. (2025). arXiv: 2506.09147.

[8] Ruhan Circi, Juanita Hicks, and Emmanuel Sikali. 2023. Automatic item generation: foundations and machine learning-based approaches for assessments. *Frontiers in Education*, Volume 8 - 2023.

[9] Bidyut Das, Mukta Majumder, Santanu Phadikar, and Arif Ahmed Sekh. 2021. Automatic question generation and answer assessment: a survey. *Research and Practice in Technology Enhanced Learning*, 16, 1, 5.

[10] Renzo Degiovanni and Jordi Cabot. 2025. Towards reliable llm-based exam generation. lessons learned and open challenges in an industrial project. In *The 40th IEEE/ACM International Conference on Automated Software Engineering (ASE) – Industry Showcase Track*.

[11] Filipe Falcão, Daniela Marques Pereira, Nuno Gonçalves, Andre De Champlain, Patrício Costa, and José Miguel Pêgo. 2023. A suggestive approach for assessing item quality, usability and validity of automatic item generation. *Advances in Health Sciences Education*, 28, 5, 1441–1465.

[12] Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. Bias and fairness in large language models: a survey. *Computational Linguistics*, 50, 3, (Sept. 2024), 1097–1179.

[13] Mark J Gierl, Okan Bulut, Qi Guo, and Xinxin Zhang. 2017. Developing, analyzing, and using distractors for multiple-choice tests in education: a comprehensive review. *Review of educational research*, 87, 6, 1082–1116.

[14] Mark J Gierl and Hollis Lai. 2013. Evaluating the quality of medical multiple-choice items created with automated processes. *Medical education*, 47, 7, 726–733.

[15] Guher Gorgun and Okan Bulut. 2024. Exploring quality criteria and evaluation methods in automated question generation: a comprehensive survey. *Education and Information Technologies*, 29, 18, 24111–24142.

[16] Jiawei Gu et al. 2025. A survey on LLM-as-a-judge. (2025). arXiv: 2411.15594.

[17] Thomas M Haladyna, Steven M Downing, and Michael C Rodriguez. 2002. A review of multiple-choice item-writing guidelines for classroom assessment. *Applied measurement in education*, 15, 3, 309–333.

[18] Lei Huang et al. 2025. A survey on hallucination in large language models: principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43, 2, Article 42, (Jan. 2025).

[19] Euigyum Kim, Salah Khalil, and Hyo Jeong Shin. 2025. Comparing human and LLM evaluations on AI-generated critical thinking items: implications for valid applications of automatic item generation. In *2nd Workshop on Automated Evaluation of Learning and Assessment Content*. Vol. 4006. CEUR-WS.org.

[20] Ghader Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. 2020. A systematic review of automatic question generation for educational purposes. *International journal of artificial intelligence in education*, 30, 1, 121–204.

[21] Patrick Lewis et al. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. (2021). arXiv: 2005.11401.

[22] Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: multimodal reasoning via thought chains for science question answering. In *The 36th Conference on Neural Information Processing Systems (NeurIPS)*.

[23] Jaclyn Martin Kowal, Kenzie Hurley Bryant, Dan Segall, and Tracy Kantrowitz. 2025. Harnessing generative AI for assessment item development: comparing AI-generated and human-authored items. *International Journal of Selection and Assessment*, 33, 3, e70021.

[24] Ruslan Mitkov, Andrea Varga, Luz Rello, et al. 2009. Semantic similarity of distractors in multiple-choice tests: extrinsic evaluation. In *Proceedings of the workshop on geometrical models of natural language semantics*, 49–56.

[25] Sergio Morales, Robert Clarisó, and Jordi Cabot. 2024. A DSL for testing LLMs for fairness and bias. In *ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems (MODELS)*. Linz, Austria.

[26] Mike Papadakis, Marinos Kintis, Jie Zhang, Yue Jia, Yves Le Traon, and Mark Harman. 2019. Chapter six - Mutation testing advances: an analysis and survey. *Advances in Computers*, 112, 275–378.

[27] Debra Pugh, André De Champlain, Mark Gierl, Hollis Lai, and Claire Touchie. 2020. Can automated item generation be used to develop high quality MCQs that assess application of knowledge? *Research and Practice in Technology Enhanced Learning*, 15, 1, 12.

[28] Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. A systematic survey of prompt engineering in large language models: techniques and applications. (2024). arXiv: 2402.07927.

[29] Sami Sarsa, Paul Denny, Arto Hellas, and Juho Leinonen. 2022. Automatic generation of programming exercises and code explanations using large language models. In *ICER 2022: ACM Conference on International Computing Education Research*. ACM, 27–43.

[30] Sergio Segura, Gordon Fraser, Ana Belén Sánchez, and Antonio Ruiz Cortés. 2016. A survey on metamorphic testing. *IEEE Trans. Software Eng.*, 42, 9, 805–824.

[31] Arjun Singh Bhatia, Manas Kirti, and Sujan Kumar Saha. 2013. Automatic generation of multiple choice questions using Wikipedia. In *International conference on pattern recognition and machine intelligence*. Springer, 733–738.

[32] Yishen Song, Junlei Du, and Qinhua Zheng. 2025. Automatic item generation for educational assessments: a systematic literature review. *Interactive Learning Environments*, 1–20.

[33] Bin Tan, Nour Armoush, Elisabetta Mazzullo, Okan Bulut, and Mark Gierl. 2024. A review of automatic item generation techniques leveraging large language models. *International Journal of Assessment Tools in Education*, 12, 2, 317–340.

[34] Aman Singh Thakur, Kartik Choudhary, Venkat Srinik Ramayapally, Sankaran Vaidyanathan, and Dieuwke Hupkes. 2025. Judging the judges: evaluating alignment and vulnerabilities in LLMs-as-judges. (2025). arXiv: 2406.12624.

[35] Lianmin Zheng et al. 2023. Judging LLM-as-a-judge with MT-bench and Chatbot Arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*. Curran Associates Inc., New Orleans, LA, USA.