LLM-Based Automatic Generation of Multiple-Choice Questions With Meaningful Distractors

Víctor Jesús Sotelo Chico[1], **André Gomes Regino** [2], **Rodrigo Bonacin** [2], **Julio Cesar dos Reis**[1]

[1]Institute of Computing – Universidade Estadual de Campinas
(UNICAMP) – Campinas – SP – Brazil

[2]Center for Information Technology Renato Archer
Campinas, SP – Brazil

v265173@dac.unicamp.br, {aregino,rbonacin}@cti.gov.br, jreis@ic.unicamp.br

Abstract. *Creating effective Multiple-Choice Questions (MCQs) with high-quality distractors that carefully evaluate students is challenging. Large Language Models (LLMs) can contribute by automatically generating questions. This study presents a novel framework for generating distractors for Portuguese-language. We evaluate the framework using Sabía-3 (Portuguese-specific) and GPT4o-mini (multilingual). We assessed the grammatical and semantic diversity of the generated distractors and devised a qualitative evaluation through Claude-3 Haiku. Results show that integrating educational principles into prompts enhances the relevance and diversity of distractors, marking progress in automated activity and assessment generation for the Portuguese language.*

1. Introduction

Online educational resources, such as e-books, videos, and audio content, support lifelong learning by providing diverse, engaging, and adaptive materials to meet learners' needs. This enables interactive and multimodal experiences that cater to different learning styles, potentially fostering more profound understanding and retention.

Evaluating students' learning in a specific topic with a wealth of information can be challenging. Assessing student knowledge is a complex task that involves evaluating both factual recall and a deeper understanding, as well as critical thinking and the ability to apply knowledge. Additionally, assessing higher-order thinking often requires more complex and subjective grading, which can lead to inconsistencies and biases in evaluation. Educators must dedicate considerable time to creating and grading evaluation activities, like questionnaires; this demands further technological support.

One potential solution is the introduction of Large Language models (LLMs) [Kumar 2024], robust neural networks capable of generating text and images, in the education field. LLMs have been utilized for personalized education support [Wen et al. 2024], Tutoring Systems [Pal Chowdhury et al. 2024], Automatic question generation [Vu et al. 2024], and other contributions related to AI for education (AI4EDU). Their application can facilitate teachers' work by reducing their efforts to create evaluations and avoiding the reuse of outdated or limited questions.

However, automatically creating MCQs involves generating alternatives with a correct answer and distractors [Awalurahman and Budi 2024]. Such distractors must re-

DOI: 10.5753/sbie.2025.12675

present plausible incorrect answers capable of challenging students at the right level. They need to assess various students' profiles without being so difficult as to confuse or frustrate learners. Distractors should be fair and unbiased, avoiding cultural, linguistic, or personal references that might favor or disadvantage certain groups of students [Gierl et al. 2017].

This article proposes and evaluates a novel framework for automatically generating adequate distractors for the Portuguese language, a topic that has been poorly investigated in the literature. Our framework aims to support learners and teachers in generating plausible distractors aligned with the educational objectives of Portuguese speakers.

Our solution approach explores prompt engineering techniques to assess how to create relevant distractors by utilizing distractors' theory across two main LLMs: gpt4o-mini [OpenAI 24], a multilingual version, and sabiá-3 [Abonizio et al. 2024], a Portuguese option. We note that our solution is independent of the LLMs (*i.e.*, model-agnostic), which allows testing and setup for other LLMs. Our study addresses the following original research questions:

1. **RQ1**: How can we integrate distractor theory into prompt instructions?
2. **RQ2**: How do automatically generated distractors compare to those created by humans?
3. **RQ3**: Are Portuguese language-focused LLMs better than a multilingual choice in generating adequate distractors?

Our experimental evaluation measures the grammatical diversity using metrics such as Self-BLEU [Zhu et al. 2018]. We introduce Self-Cosine to measure semantic sharing between the distractors. We set up a Claude-3 Haiku LLM as a judge evaluator using the item Writing Flaw Theory (IWFT) [Moore et al. 2023], considering its guidelines for evaluation. To the best of our knowledge, we are the first to apply this approach to distractor generation, specifically in the context of Portuguese texts. All codes running our solution and experiments are available in an public repository[1]. Our research demonstrates that integrating educational theory related to distractors into prompt instructions for LLMs improves the generation of distractors, making them more relevant and contextually appropriate and aligning better with educational objectives.

The remainder of this article is organized as follows: Section 2 presents the relevant concepts. Section 3 offers a synthesis and analysis of key related studies. Section 4 details our proposed methods for generating distractors in the Portuguese language. Section 5 outlines our experimental evaluation. Section 6 discusses our findings. Finally, Section 7 summarizes the conclusions and suggests future research directions.

2. Background

Multiple-choice questions (MCQs) are a type of assessment in which a person is asked to identify the correct answer from a list of possible options [CH and Saha 2020]. MCQs are composed of three main elements:

- **Stem** represents a question or problem to be answered.
- **Key** indicates the correct answer to the stem.
- **Distractors** is a list of elements of plausible and incorrect answers to the stem.

[1] `https://github.com/Studyard/Distractors`

We present an example of MCQ indicating each element:

What is the primary reason the Earth experiences different seasons? (Stem)

A) The Earth is closer to the Sun in summer and farther away in winter. (Distractor).
B) The Earth's axis tilt causes varying sunlight angles throughout the year. **(Key)**.
C) The Earth's speed in its orbit around the Sun changes. (Distractor).
D) The Moon's gravitational pull influences seasonal changes. (Distractor)

Effective distractors should offer complex properties to challenge students. This study refers to these properties as Distractors Theory (DT) [Gierl et al. 2017]. We further elaborate on specific aspects of Distractor Theory. For instance, *effective distractors should be similar in length*, as significant discrepancies can attract undue attention or lead students to discard specific options prematurely. They should be *mutually exclusive*, ensuring no meaning overlap confuses the test-taker. Moreover, distractors must be *grammatically consistent with the stem* to maintain syntactic coherence. Finally, they should *avoid absolute terms* such as "always", "never", "all" or "none" because such options are often easily dismissed by students.

Item Writing Flaws (IWFs) are violations of the MCQ construction that can influence students' performance, making the items overly complex or too simplistic [Rush et al. 2016]. The IWF uses rubrics about how each element in MCQ (stem, key, and distractors) must behave, including those described for distractors. For instance, the MCQ presented in the given example contains an IWF, which can reduce its validity. It includes misleading distractors (*e.g.*, Option C– orbital speed affects seasons) and an irrelevant option (Option D – Moon's gravity). It reinforces a common misconception (Option A – Earth's distance from the Sun causes seasons). These flaws introduce ambiguity and potential misconceptions, making it difficult to assess students' understanding.

Large Language Models (LLMs) are artificial neural networks that generate natural language text, code, visual question answers, and document embeddings [Kumar 2024]. They are trained with massive amounts of data using unsupervised techniques. They suffer from some downsides, including an LLM's ability to sense information, known as hallucination [Perković et al. 2024]. Prompt engineering is a technique to refine a prompt [Marvin et al. 2024], concerning adequate instructions to direct an LLM output. Such instructions must enable an LLM to follow the process for the desired outcomes.

BLEU metric is commonly used for automatic machine translation, in which the generated translation is compared with a set of references [Papineni et al. 2002]. Higher values indicate a favorable translation. BLEU can help measure the diversity between Key (reference or ground truth) and $Distractors$ (predicted values) by assessing how similar the distractors are in grammar to the Key. A lower BLEU score indicates higher diversity as the distractors differ from the key. Self-BLEU is a metric based on the BLUE metric to measure diversity between a set of values $(v_1, v_2, .., v_n)$ without depending on any ground truth [Zhu et al. 2018]. This makes it suitable for application in LLM generations, as in most cases, an LLM does not have a reference to compare its output. The value tends to be high when identical distractors are presented, helping to identify them. However, if the distractors vary, the value decreases. According to the IWF principles, identical distractors are unsuitable. Our study also explores the Cosine of Similarity as a metric that compares vectors of the same length by applying a *cosine* between them

[Shree Charran et al. 2022]. Such metrics become essential due to the embedding models, as their representation is presented as numeric vectors, and similar vectors represent sentences with similar semantic levels (*i.e.*, those that are close in meaning).

3. Related Work

Ren and Zhu [Ren and Q. Zhu 2021] proposed an automated framework for generating distractors for English cloze-style multiple-choice questions (MCQs). They utilized a knowledge base to produce potential distractors and selected the most suitable one through techniques based on morphological similarity and contextual embedding. The study achieved plausibility and diversity in its distractors. A human evaluation was conducted to validate the results using the F1-score measure to determine whether the generated distractors matched the ground truth.

Cavusoglu *et al.* [Çavuşoğlu et al. 2024] proposed a solution that utilizes Pre-trained Language Models and Masked Prediction learning. The solution masks a key element from a source text to create a fill-in-the-blank question and employs the mask prediction task to generate distractors. They eliminated those that are similar to the key answer as well as those that are similar to each other. ChatGPT selected the best distractors, and 30 participants tested and scored the fill-in-the-blank questions. The results confirmed that the generated distractors were adequate.

Yu *et al.* [Yu et al. 2024] proposed to modify the pre-trained process of language models to enhance distractor generation. The authors tested their proposal using BART and T5 models, where the elements of MCQs are used to construct a set of pseudo-questions. For example, a key is used in an MCQ to recover text from a dataset such as Wikipedia. The key is masked in the recovered text, generating a pseudo-question. Then, the model is trained with pseudo questions and distractors to create plausible distractors.

Rodriguez *et al.* [Rodriguez-Torrealba et al. 2022] proposed an end-to-end system for generating MCQs, and T5 models were used to train for text-to-text generation. The process was organized into two parts: first, they fine-tuned a model specifically to generate question-answer (QA) pairs; then, they fine-tuned another model to create distractors using the QA pairs as input. They achieved strong ROUGE metrics and validated the generated questions through human evaluation for formatting and grammar.

Wang *et al.* [Wang et al. 2023] focused on the fill-in-the-blank task to elaborate the *VocaTT* (vocabulary teaching and training) solution. Their solution includes preprocessing, sentence generation (stems), and creation of candidate word options using GPT. The final step generates a set of candidate distractors and selects the most suitable ones. A human evaluation of 60 automatically generated distractors shows that the most generated sentences were relevant, and the word distractors were suitable.

Gonçalo *et al.* [Gonçalo Oliveira et al. 2023] proposed several methods for generating distractors in the Portuguese language. The primary strategies involved using context to identify entities similar to the original responses, leveraging resources such as *DBpedia* and *WordNet*, and applying semantic similarity to produce plausible distractors. Distractors were created for numerical responses by generating values within a specified range. BERTimbau [Souza et al. 2020] and GPT-2 [Radford et al. 2019] were used as distractor classifiers. Human evaluation was used to assess distractor quality.

Chico *et al.* [Chico et al. 2024] proposed the *BEQuizzer* framework to create MCQs. The authors performed a comparative analysis between pre-trained language models and LLMs by using the EXAMS datasets. They proposed the use of fine-tuning language models and prompt engineering to generate the elements of MCQ in a unique step. The LLM performed better at creating adequate stems, keys, and distractors. They used NILC-Metrix to evaluate the quality of the synthetic generating texts.

Our present study significantly advances the state-of-the-art of distractors generation for the Portuguese language. Unlike existing studies, our approach does not require massive data to fine-tune pre-trained models. We introduce novel methods for automatic distractor generation rooted in prompt engineering and their integration with elements of education theory. We propose automatic metrics to measure diversity, as existing metrics used to evaluate MCQ generation have limitations. In our experimental assessment, we utilized an LLM to assess the quality of distractors, aligning with educational guidelines (IWF) for constructing MCQs.

4. Automatic Generation of MCQs

The automatic generation of MCQs involves creating three essential components: the Stem, the Key, and the Distractors. Figure 1 presents our overall method for generating MCQs. We follow a two-step process to generate MCQs: 1) creating the Stem and Key, and 2) developing the Distractors. Figure 1 presents the parameters provided by a user, colored in orange, while the elements generated by an LLM are shown in purple. Finally, the blue box represents specific prompts and instructions passed to the distractor generators.

In the first step, an LLM is asked to generate a pair *Stem* and *Key* elements, receiving an input document (Doc) as a parameter, which represents a knowledge base used to create the MCQ. The LLM box includes a set with an exclusively adjusted prompt to generate only a Stem and Key.

In the second step, the generated Stem and Key are passed to another specific prompt with instructions about generating the distractors. The parameter n is passed along with the generated elements to indicate the number of required distractors. The second step may use the same LLM-A as the first or a different LLM-B for distractors.

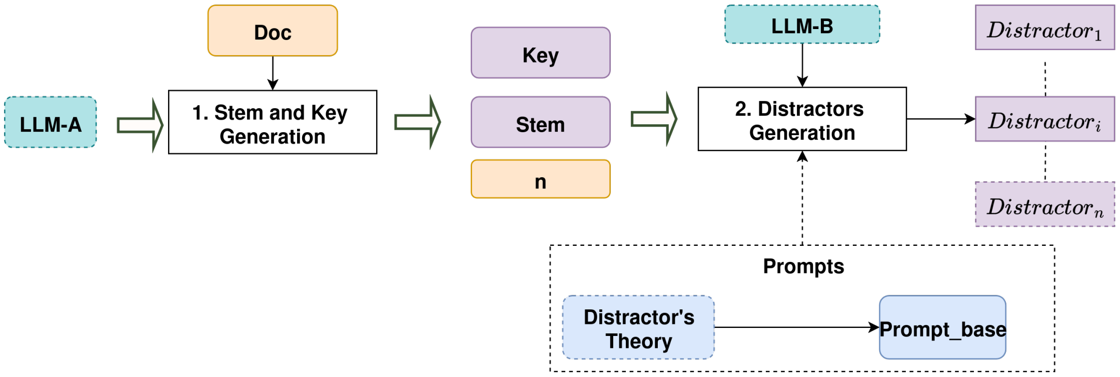

Figure 1. Method for automatically generating MCQs

Our proposed pipeline for generating distractors uses the *Key* and *Stem* generated from the first step (cf. Figure 1). They are input along with the document doc, which serves as a content base to generate n distractors automatically.

Such parameters are passed to our module for distractor generation. It contains prompt-based instructions to create distractors. These instructions lacked educational objectives to build the distractors. Our approach injects information into the prompt to enrich their knowledge about what distractors must be generated. The union of both prompts (prompt base + Distractor theory, cf. Figure 1) was passed to the LLM used for creating distractors.

5. Experimental Evaluation

This section presents the evaluation conducted to assess the effectiveness of our proposal. Our evaluation considers two LLMs to generate distractors. We developed two prompt templates for generating plausible distractors: the base form (BF) and a Distractor Theory version (DT)[2]. The rationale is to show how distractors' theoretical knowledge affects performance compared to a baseline prompt instruction.

We present the overall procedures (Subsection 5.1) followed by a description of the datasets (Subsection 5.2), language models (Subsection 5.3), and metrics (Subsection 5.4). Subsection 5.5 reports on the obtained experimental results.

5.1. Procedures

Figure 2 presents an overview of the evaluation pipeline, organized into four main steps (represented by orange rectangles).

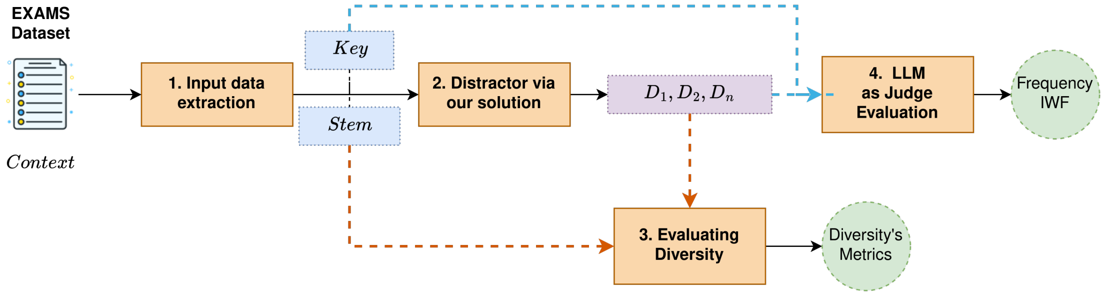

Figure 2. Procedure for evaluating our automatically generated distractors, measuring diversity, and using LLM as a Judge for assessing IWF presence

1. Data input extraction. This step extracts relevant information from the dataset (cf. Subsection 5.2). We focus on gathering the *stem*, *answer*, and related *context*. The context refers to the dataset's metadata, which was used to elaborate on the MCQs.

2. Distractor generation. This step receives the extracted data from the *input data extraction* (Key, $Stem$, $Context$). We used various setups with distinct LLMs (cf. Section 5.3) and prompts, obtaining a set of distractors (d_1, d_2, d_n).

3. Evaluating diversity metrics. After generating the distractors, this step computes their diversity. This involves comparing one generated distractor with another. We

[2] Details of these prompts are available in our repository.

assess how these distractors align with the reference answers from the datasets by calculating the similarity metrics. Subsection 5.4 provides further details on how to compute and define these metrics.

4. LLM as a Judge evaluation. It presents our pipeline for evaluating the presence of IWF. The first step provides the *Stem* and *Key* to the distractor generator. Next, the LLM generates distractors, combined with the *Stem* and *Key* from the dataset. This complete set is submitted to the LLM judger utilizing the Claude-3-haiku model. This prompt is equipped with the criteria to identify IWFs. It checks the presence of IWFs in the content. The model produces a Boolean value ("Status"). It returns *true* if a match is found according to the defined IWF criteria, indicating that the evaluator considers no existing issues with that IWF. If no match is found, it returns *false*. Such an evaluation provides us with initial feedback on whether the distractors are effective. Additionally, the output provides a text explanation labeled "Reason", which details the rationale behind the model's response.

5.2. Datasets

We chose the EXAMS dataset [Hardalov et al. 2020] because it includes high school exam questions in 16 languages, including our target language, Portuguese, and covers multiple types of questions, such as fill-in-the-blanks, True/false, and matching. We decided to filter a base form of the MCQ, removing fill-in-the-blank questions. We aimed to reduce the influence of more specific question types and create a homogeneous dataset. We filtered the questions by eliminating the fill-in-the-blanks in the validation split and reduced the split to 159 examples.

5.3. Language Models

We choose two *LLMs as distractor generators*: *GPT-4o-mini*[3] and *Sábia-3* [Abonizio et al. 2024], multilingual and Portuguese model, respectively. We used them to evaluate their ability to create adequate distractors for Portuguese MCQs. Our rationale is that GPT models are currently state-of-the-art and support several languages, including Portuguese. Sabiá has emerged as a leading model for Portuguese applications.

For the *LLM as an evaluator* task, we used *Claude-3 Haiku*[4]. Our choice relies on using an LLM as a Judge [Zheng et al. 2023] to evaluate the effectiveness of the results generated by other models of different providers (OpenAI, Maritaca). Our decision aims to prevent an LLM model from grading its responses in a biased manner [Xu et al. 2024] and ensure the evaluator model comes from a different provider.

5.4. Evaluation Metrics

Figure 3 presents the taxonomy of our metrics, organized into two parts: *Diversity* and *Frequency of the IWF*. The diversity metrics are categorized into reference-free metrics and reference-based metrics.

I) Diversity automatic metrics. Diversity is one key criterion for assessing the quality of the distractor generator. One of the primary properties of distractors is their distinctiveness from one another. This is important because closely related distractors can

[3] https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/
[4] https://www.anthropic.com/news/claude-3-family

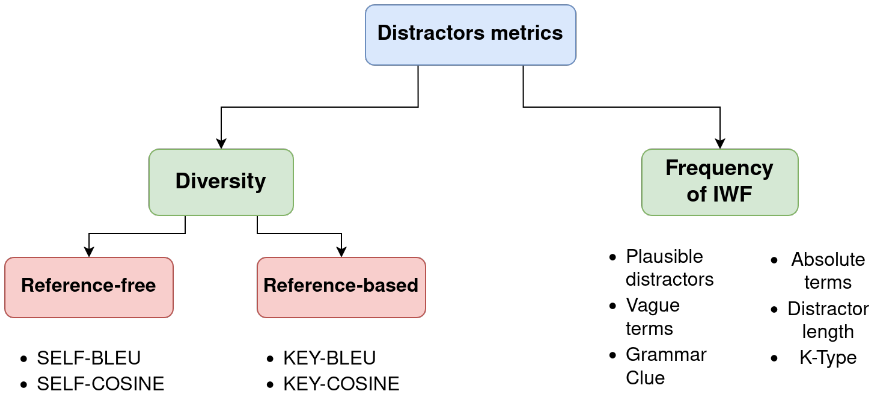

Figure 3. Taxonomy of the metrics for evaluating distractors generation

be easily dismissed by students immediately. We proposed using automatic metrics to measure the diversity between our generated distractors.

I.1) Reference-based metrics. The dataset used as input comprises complete MCQs. This means it already contains accurate information for the correct answers and the expected distractors. We proposed using this data to compute specific measures.

KEY-BLEU and KEY-COSINE. First, we propose using two metrics to determine how our distractors are similar to the key (correct answer). We compare the key to each n-generated distractor (d_i) using a similarity score. Then, we compute the mean of each score. BLEU and cosine were used as similarity scores measuring grammar and semantic similarity, respectively. We refer to the proposed metrics as *KEY-BLEU* and *KEY-COSINE*, respectively.

I.2) Reference-free metrics. These metrics are used to measure the similarity of our distractors (only) regarding grammar and meaning sharing.

SELF-BLEU. We compute the SELF-BLEU [Zhu et al. 2018] metric for grammar similarity between all distractors, taking one distractor as predicted values and the others as references to compute BLEU. We repeated the process to take the others as expected. Finally, we calculate the mean of their BLEU scores. This metric indicates whether the distractors shared grammar similarities.

SELF-COSINE: Taking inspiration from SELF-BLEU, we propose the metric ***SELF-COSINE***. Our SELF-COSINE metric is defined as the average pairwise cosine similarity among all document embeddings. Specifically, it corresponds to the mean of the values in the upper triangular portion of the similarity matrix (excluding the diagonal).

II) Frequency of IWFs. In addition to the diversity, we measure the frequency of IWFs. This is relevant to understanding how the application of the distractors' theory influences distraction quality. Table 1 summarizes our selected criteria for our chosen IWF. We focused only on the IWF, which addresses the assessment of distractors.

We created a modification from the existing writing flaw ***Longest responses are always the correct*** to evaluate if the generated distractors are similar in length to the correct answer; this modified IWF is relevant in the context of LLM, where the model tends

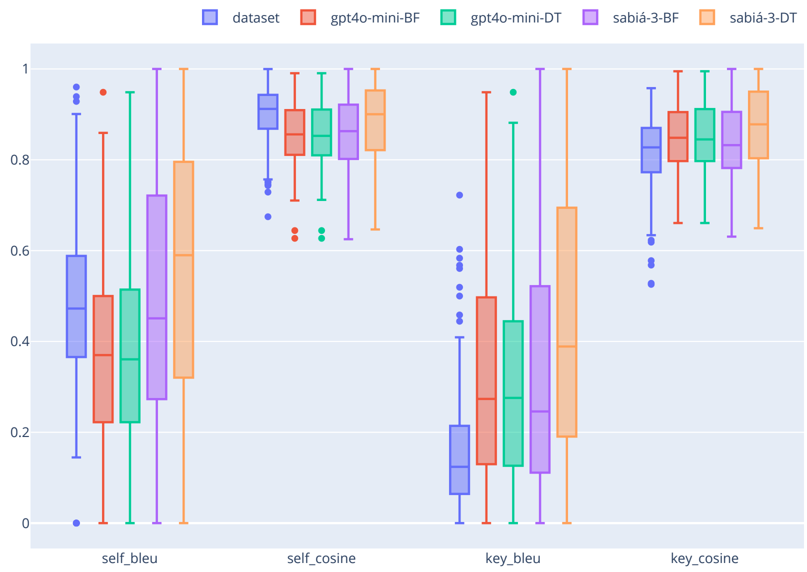

Figure 4. Boxplot comparing the distribution of reference-free metrics (SELF-BLEU and SELF-COSINE) and reference-based metrics (KEY-BLEU and KEY-COSINE) across five evaluation cases.

to generate the most extended sequences. All distractors must meet specific conditions to be scored as "good distractors" (scored as one by the LLM). As this metric counts the occurrence of a condition (whether the LLM identifies that it does not exist IWF), a higher value indicates better quality in the distractors across the overall datasets, with a maximum score of 159 (total examples in the dataset).

Table 1. Item Writing Flaws criteria for our LLM as judge

IWF	Criteria
Plausible distractors	All distractors are plausible
Vague terms	All distractors should avoid using vague terms
Grammar Clue	All options should be grammatically consistent with the stem
Absolute terms	All distractors must avoid the use of absolute terms
Distractor length	All distractors must be similar in size to the correct answer.
K-Type	All distractors must avoid containing combinations of other response option

5.5. Evaluation Results

Diversity Evaluation Results. Figure 4 presents the boxplot of the distribution of the diversity scores for the original dataset, gpt4o-mini-BF, gpt4o-mini-DT, Sabiá-3-BF, and Sabiá-3-DT. This Figure shows that the MCQ elements present in the original dataset present more outliers in their distribution of similarity scores.

The Sabiá-3-DT model exhibits the broadest range of values across the metrics, particularly in self_bleu and key_bleu. In contrast, the dataset baseline exhibits high median values in self_bleu and self_cosine, but shows lower variation in key_cosine. The gpt4o-mini-BF and gpt4o-mini-DT models presented more compact distributions with minimal variation across most metrics. All models exhibited closely clustered values in key_cosine, with gpt4o-mini-BF and gpt4o-mini-DT showing tight distributions.

Table 2 presents the results applying automatic similarity metrics (cf. Section 5.4); over this table, we present two language models, Sabiá-3 and GPT4o-mini. Regarding grammar similarity, we observed that GPT4o-mini using DT achieved more diversity (lower Self-BLEU values) compared to the other configurations, which computed only the generated distractors. When using the answer as reference (Key-BLEU) to compare the generated distractors, lower values were obtained for the distractors presented in the original EXAMS dataset, followed by the GPT4o-mini using the prompt in its BF.

Regarding semantic metrics, Sabiá-3 with the distractor theory achieved the highest similarity values for Self-Cosine, which were close to those of the distractors in the EXAMS dataset. When comparing the semantics of the distractors to the key (Key-Cosine), Sabiá-3 obtained the highest semantic score, while the other LLMs achieved similar lower scores. Meanwhile, the data from the datasets achieved the lowest score.

Table 2. Results of diversity evaluation over the four configurations in our experiments and the original datasets distributions.

Setup	Self BLEU	Key BLEU	Self Cosine	Key Cosine
Dataset	0.4841	**0.1590**	0.8997	0.8130
GPT4o-mini BF	0.3848	0.3136	0.8552	0.8478
GPT4o-mini DT	**0.3809**	0.3152	0.8547	0.8471
Sabiá-3 BF	0.4789	0.3370	0.8595	0.8411
Sabiá-3 DT	0.5535	0.4299	**0.8833**	**0.8711**

Evaluation Results regarding Frequency of Item Writing Flaws. Table 3 presents the results of using the Claude-3 Haiku as a judge based on the given criteria of IWF (cf. Table 1). This evaluator analyzed every generation of each model, comparing results with and without Distractor Theory in the prompts. We computed the values and compared them with the EXAMS dataset.

Plausible distractors' identification obtains a high score for almost any setup (higher than 97 incidents); such results are similar to those in the EXAMS dataset. Only the GPT4o-mini DT got a lower number compared to the baseline dataset. We noticed that by changing the prompt strategy of Sabiá-3, we found no considerable difference, while DT reduces the number of well-plausible distractors for GPT4o-mini; The GPT4o-mini DT achieved a higher score for *Vague terms* and had similar values to the EXAMS dataset. Applying DT in Sabiá-3 increases the score.

In *Grammar clue*, the highest value belongs to the EXAMS dataset, followed by GPT4o-mini BF, while the other models got similar lower values. The application of DT does not affect GPT4o-mini and Sabiá-3.

We found that Sabiá-3-BF achieved better results by avoiding the use of *Absolute terms* compared to other LLM setups. In this IWF, adding DT to the prompt did not improve the frequency for any of the evaluated models. When comparing the dimension of the distractors (*Distractors length*) with the answer, applying BF configuration positively affected both Sabiá-3 and GPT4o-mini. These values are similar to the EXAMS dataset.

Sabiá-3-DT achieved the highest *K-Type* score. We noticed that DF only had a positive impact on Sábia-3.

Table 3. Result of using Claude-3 Hayku as LLM judge using six IWF criteria to evaluate the quality of the distractors generation produced; The table shows when a distractors generation fit the IWF criteria of good distractors for each example (Total of examples 159)

	Plausible distractors	Vague terms	Grammar Clue	Absolute terms	Distractor length	K-Type
Dataset	98	76	**152**	107	85	54
GPT4o-mini BF	97	72	151	106	84	56
GPT4o-mini DT	86	**78**	147	98	**87**	55
Sabiá-3 BF	98	53	145	***112***	79	52
Sabiá-3 DT	**99**	69	145	100	**87**	**61**

6. Discussion

The Sabiá-3-DT model showed the broadest range of metric values, particularly in self-BLEU and key-BLEU, indicating high variability and diverse outputs. In contrast, the EXAMS dataset, which serves as the baseline, presented a high median in self-BLEU and self-cosine scores with less variability, indicating stable effectiveness but lower diversity.

The gpt4o-mini-BF and gpt4o-mini-DT models showed further compacted distributions, indicating consistent outputs. Key-cosine values are tightly clustered across all models, with the gpt4o-mini variations exhibiting minimal spread in key term usage.

Results in Table 2 revealed that the distractors generated by the OpenAI-based LLM are more diverse (Self-BLEU) than those found in the EXAMS dataset. The Sabiá model, across both configurations, showed less diversity, suggesting that the distractors exhibit shared grammatical structures. GPT4o-mini showed superior grammatical diversity concerning the key (correct choice). This indicates that these models may be capable of generating distractors that would not be easily discarded by students in a real-world test setting (K-type, cf. Table 1). At the same time, the cosine values indicated that the generated distractors for reference and free-based metrics (Self and Key Cosine) preserved the semantics of the distractors across the tested dataset.

Using an LLM as a judge to evaluate whether the distractors' generation fit into IWF criteria for high-quality distractors provided us with another perspective. According to the judge model, every model achieved more than 86 plausible distractors. This indicates that the judge model evaluator failed to recognize that all the generated distractors were, for instance, good enough as distractors. We noticed that our automatically generated distractors were judged to be similar in quality to the existing distractors in the EXAMS dataset (human-created). This is evidence of the potential of our solution to be used in the educational field.

Detecting vague and absolute terms exhibits different behaviors. Vague terms show slight improvement when a distractor theory is introduced. However, because vague terms may not be clearly defined in the criteria (cf. Table 1), this limitation allows for a broader interpretation of what constitutes a vague term. In contrast, absolute terms are more closely related and clearly defined, which helps the LLM evaluator identify them. This explains why evaluating absolute terms tends to yield higher initial values. Besides, the introduction of the distractor theory tends to improve the performance in both LLMs.

In our experimental evaluation of the Grammar Clue, our distractors' generator achieved at least 91.19%. This suggests our distractors align well with the stem, reducing the risk of hallucinated options that students could easily spot.

Comparing the size of distractors with the related correct answers yielded results comparable to those in the EXAMS dataset. The values still account for only about 50% of the total examples. This indicates that the model is generating distractors of varying lengths. This could pose a problem for automatic MCQs, as learners can notice these differences and quickly identify the correct answer.

The last evaluation criterion, the K-type, is essential because some distractors may be combinations of other options. This situation affects the principle of creating effective MCQs. In such cases, repeated exposure can help students who may not initially know the correct answer to the stem, allowing them to arrive at the right choice by eliminating options based on their characteristics. Analyzing results in Table 2 and Table 3, we noticed that K-type is lower for Sabiá-3 model, explained by their higher Self-BLEU. This means we can detect distractors' combinations (K-type) by computing the SELF-BLEU because this tends to be high for grammar sharing.

We found that a Portuguese-focused LLM was the most effective in generating suitable distractors according to the judge model. In addition, our solution of introducing more specific theories of distractors proved beneficial for three IWF criteria (Vague Terms, Distractor Length, and K-type – cf. Table 3).

This study found that incorporating the distractor theory elements into the prompt, with each element marked by a tag, improved the quality of the distractors (answering question RQ1). The output from the LLM exhibited similar properties to those in the baseline dataset, and there was no increase in outliers (answering question RQ2). Both evaluated models improved their IWF frequency values. The Sabiá-3 model aligned better and scored higher (answering question RQ3).

7. Conclusion

Creating high-quality distractors is an open research challenge. Our investigation proposed a novel alternative using prompt engineering based on Distractor Theory, significantly improving the quality of the generated distractors. This approach enhanced the automatic generation of MCQs. Our research extended the current knowledge methods for evaluating MCQs. This study introduced a set of metrics designed to measure diversity in Portuguese language-generated MCQs using pre-trained language models. We assessed a first-ever LLM evaluator tailored for Portuguese MCQs based on IWF criteria. Future studies aim to develop an overall end-to-end solution for automatically generating and evaluating MCQs by extending our LLM-judge evaluator to incorporate additional IWF criteria into the evaluation.

Ethical Considerations

This study used only publicly available datasets and did not involve any human participants. All analyses were conducted using large language models (LLMs) without collecting or processing personal or sensitive data. Since no human subjects were involved and all data were openly accessible, ethical approval from a Research Ethics Committee was not required. The study followed all applicable institutional and academic guidelines for research using public data and artificial intelligence tools.

Acknowledgments

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. We also thank the National Council of Technological and Scientific Development (CNPq), Brazil, grant #301337/2025-0.

Referências

Abonizio, H., Almeida, T. S., Laitz, T., Junior, R. M., Bonás, G. K., Nogueira, R., and Pires, R. (2024). Sabiá-3 technical report.

Awalurahman, H. W. and Budi, I. (2024). Automatic distractor generation in multiple-choice questions: a systematic literature review. *PeerJ Computer Science*, 10:e2441.

Çavuşoğlu, D., Şen, S., and Sert, U. (2024). DisGeM: Distractor generation for multiple choice questions with span masking. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N., editors, *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9714–9732, Miami, Florida, USA. Association for Computational Linguistics.

CH, D. R. and Saha, S. K. (2020). Automatic multiple choice question generation from text: A survey. *IEEE Transactions on Learning Technologies*, 13(1):14–25.

Chico, V. J. S., Tessler, J. F., Bonacin, R., and dos Reis, J. C. (2024). Bequizzer: Ai-based quiz automatic generation in the portuguese language. In Rapp, A., Di Caro, L., Meziane, F., and Sugumaran, V., editors, *Natural Language Processing and Information Systems*, pages 237–248, Cham. Springer Nature Switzerland.

Gierl, M. J., Bulut, O., Guo, Q., and Zhang, X. (2017). Developing, analyzing, and using distractors for multiple-choice tests in education: A comprehensive review. *Review of Educational Research*, 87(6):1082–1116.

Gonçalo Oliveira, H., Caetano, I., Matos, R., and Amaro, H. (2023). Generating and Ranking Distractors for Multiple-Choice Questions in Portuguese. In Simões, A., Berón, M. M., and Portela, F., editors, *12th Symposium on Languages, Applications and Technologies (SLATE 2023)*, volume 113 of *Open Access Series in Informatics (OASIcs)*, pages 4:1–4:9, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

Hardalov, M., Mihaylov, T., Zlatkova, D., Dinkov, Y., Koychev, I., and Nakov, P. (2020). EXAMS: A multi-subject high school examinations dataset for cross-lingual and multilingual question answering. In Webber, B., Cohn, T., He, Y., and Liu, Y., editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5427–5444, Online. ACL.

Kumar, P. (2024). Large language models (llms): survey, technical frameworks, and future challenges. *Artificial Intelligence Review*, 57(10):260.

Marvin, G., Hellen, N., Jjingo, D., and Nakatumba-Nabende, J. (2024). Prompt engineering in large language models. In Jacob, I. J., Piramuthu, S., and Falkowski-Gilski, P., editors, *Data Intelligence and Cognitive Informatics*, pages 387–402, Singapore. Springer Nature Singapore.

Moore, S., Fang, E., Nguyen, H. A., and Stamper, J. (2023). Crowdsourcing the evaluation of multiple-choice questions using item-writing flaws and bloom's taxonomy.

In *Proceedings of the Tenth ACM Conference on Learning @ Scale*, L@S '23, page 25–34, New York, NY, USA. Association for Computing Machinery.

OpenAI (24). GPT-4o mini: advancing cost-efficient intelligence.

Pal Chowdhury, S., Zouhar, V., and Sachan, M. (2024). Autotutor meets large language models: A language model tutor with rich pedagogy and guardrails. In *Proceedings of the Eleventh ACM Conference on Learning @ Scale*, L@S '24, page 5–15, New York, NY, USA. Association for Computing Machinery.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Perković, G., Drobnjak, A., and Botički, I. (2024). Hallucinations in llms: Understanding and addressing challenges. In *2024 47th MIPRO ICT and Electronics Convention (MIPRO)*, pages 2084–2088.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners.

Ren, S. and Q. Zhu, K. (2021). Knowledge-driven distractor generation for cloze-style multiple choice questions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(5):4339–4347.

Rodriguez-Torrealba, R., Garcia-Lopez, E., and Garcia-Cabot, A. (2022). End-to-end generation of multiple-choice questions using text-to-text transfer transformer models. *Expert Systems with Applications*, 208:118258.

Rush, B. R., Rankin, D. C., and White, B. J. (2016). The impact of item-writing flaws and item complexity on examination item difficulty and discrimination value. *BMC Medical Education*, 16(1):250.

Shree Charran, R., Dubey, R. K., and Jain, S. (2022). Chapter 18 - chronological text similarity with pretrained embedding and edit distance. In Pandey, R., Khatri, S. K., kumar Singh, N., and Verma, P., editors, *Artificial Intelligence and Machine Learning for EDGE Computing*, pages 279–286. Academic Press.

Souza, F., Nogueira, R., and Lotufo, R. (2020). Bertimbau: Pretrained bert models for brazilian portuguese. In *Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, October 20–23, 2020, Proceedings, Part I*, page 403–417, Berlin, Heidelberg. Springer-Verlag.

Vu, S. T., Truong, H. T., Do, O. T., Le, T. A., and Mai, T. T. (2024). A chatgpt-based approach for questions generation in higher education. In *Proceedings of the 1st ACM Workshop on AI-Powered QA Systems for Multimedia*, AIQAM '24, page 13–18, New York, NY, USA. Association for Computing Machinery.

Wang, Q., Rose, R., Orita, N., and Sugawara, A. (2023). Automated generation of multiple-choice cloze questions for assessing English vocabulary using GPT-turbo 3.5. In Hämäläinen, M., Öhman, E., Pirinen, F., Alnajjar, K., Miyagawa, S., Bizzoni, Y., Partanen, N., and Rueter, J., editors, *Joint 3rd International Conference on Natural*

Language Processing for Digital Humanities and 8th International Workshop on Computational Linguistics for Uralic Languages, pages 52–61. ACL.

Wen, Q., Liang, J., Sierra, C., Luckin, R., Tong, R., Liu, Z., Cui, P., and Tang, J. (2024). Ai for education (ai4edu): Advancing personalized education with llm and adaptive learning. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, page 6743–6744, New York, NY, USA. Association for Computing Machinery.

Xu, W., Zhu, G., Zhao, X., Pan, L., Li, L., and Wang, W. (2024). Pride and prejudice: LLM amplifies self-bias in self-refinement. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15474–15492, Bangkok, Thailand. Association for Computational Linguistics.

Yu, H. C., Shih, Y. A., Law, K. M., Hsieh, K., Cheng, Y. C., Ho, H. C., Lin, Z. A., Hsu, W.-C., and Fan, Y.-C. (2024). Enhancing distractor generation for multiple-choice questions with retrieval augmented pretraining and knowledge graph integration. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11019–11029, Bangkok, Thailand. Association for Computational Linguistics.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E., Zhang, H., Gonzalez, J. E., and Stoica, I. (2023). Judging llm-as-a-judge with mt-bench and chatbot arena. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S., editors, *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc.

Zhu, Y., Lu, S., Zheng, L., Guo, J., Zhang, W., Wang, J., and Yu, Y. (2018). Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1097–1100, New York, NY, USA. Association for Computing Machinery.