

# Distractor generation for multiple-choice questions with predictive prompting and large language models

Semere Kiros Bitew, Johannes Deleu,  
Chris Develder, and Thomas Demeester

IDLab, Ghent University – imec, Ghent, Belgium  
{semerekiros.bitew, johannes.deleu, chris.develder,  
thomas.demeester}@ugent.be

**Abstract.** Large Language Models (LLMs) such as ChatGPT have demonstrated remarkable performance across various tasks and have garnered significant attention from both researchers and practitioners. However, in an educational context, we still observe a performance gap in generating distractors — i.e., plausible yet incorrect answers — with LLMs for multiple-choice questions (MCQs). In this study, we propose a strategy for guiding LLMs such as ChatGPT, in generating relevant distractors by prompting them with question items automatically retrieved from a question bank as well-chosen in-context examples. We evaluate our LLM-based solutions using a quantitative assessment on an existing test set, as well as through quality annotations by human experts, i.e., teachers. We found that on average 53% of the generated distractors presented to the teachers were rated as high-quality, i.e., suitable for immediate use as is, outperforming the state-of-the-art model. We also show the gains of our approach<sup>1</sup> in generating high-quality distractors by comparing it with a zero-shot ChatGPT and a few-shot ChatGPT prompted with static examples.

**Keywords:** Distractor generation · natural language processing · large language models · predictive prompting · language learning · neural networks.

## 1 Introduction

The rapid advancement in artificial intelligence (AI) and large language models (LLMs) have paved the way for transformative applications across various domains, including the education domain. Since several LLMs (e.g., GPT-3 [4], InstructGPT [23], GPT-4 [22]) have been pretrained on massive amounts of data across multiple domains and languages, they are capable of solving natural language processing (NLP) tasks with little training examples (i.e., few-shot learning) or no additional training (i.e., zero-shot learning). This opens up new

---

<sup>1</sup> <https://github.com/semerekiros/distractGPT/>

opportunities for adopting LLMs in the development of many educational technological solutions that aim to automate time-consuming and laborious educational tasks such as generating questions [15] and exercises [2], essay scoring [28], and automated feedback [5].

In particular, the recent release of ChatGPT, an LLMs-based generative AI model that requires only natural language prompts without additional model training or fine-tuning, has demonstrated diverse potential in automating various educational tasks. For example, ChatGPT has achieved the equivalent of a passing score for a third-year medical student (above 60%) in the United States Medical Licence Examination (USMLE) Step 1 exam, and provided logical justification and informational context across the majority of answers [11]. Likewise, ChatGPT’s performance on four real exams (containing 95 MCQs and 12 essay writing questions), at the University of Minnesota Law School was equivalent to C+ students implying a pass in the course [6]. Li *et al.* [17] show the capability of ChatGPT in generating high-quality reflective responses in writing assignments administered for pharmacy courses.

One important educational task is the generation of multiple-choice questions (MCQs). MCQs have long been a popular form of formative and summative assessment in education due to their automatic scoring capability and the potential they hold for delivering timely and targeted feedback, which is crucial for facilitating effective learning [29]. However, the process of crafting high-quality MCQs with effective distractors (i.e., plausible yet incorrect answers) has traditionally been both a challenging and time-consuming task for educators (e.g., teachers, content creators etc. ) as poorly prepared distractors undermine the quality of MCQs [10]. This is where LLMs offer substantial benefits as they can be leveraged to automate the MCQ construction process, thus saving educators’ time and effort while maintaining the quality and validity of the assessment items. For instance, teachers could employ LLMs to not only create different variants of the same MCQ questions but also develop different MCQs of comparable difficulty levels, facilitating targeted assessment for students with similar proficiency levels. Furthermore, students can benefit from the availability of several MCQs, enabling them to engage in regular practice, which is a well-established and highly effective learning strategy [31]. Additionally, such models could be used for large-scale testing contexts (e.g., licensure and certification testing) in which it is necessary to have multiple forms of a test and to introduce new question items regularly to minimize security concerns related to item exposure.

In a recent study [3] conducted around the same time as the release of ChatGPT, researchers used local language models to automatically retrieve and reuse distractors to create new MCQs for education by leveraging existing pools of question items. In a user study they conducted with teachers, 3 out of 10 distractors proposed by their system were found to be high-quality, which is generally sufficient for creating an MCQ, as an average MCQ typically contains 3 distractors. However, they also report a staggering 50% production of distractors that were entirely out of context given a question (so-called “nonsense distractors”). With the emergence of ChatGPT, the question arises: *does this previous*

*approach become obsolete?* In our current study, we aim to address this question by examining the ability of out-of-the-box ChatGPT to generate effective distractors to be measured on the same scale as the previous study and evaluated by experts. Moreover, we study how both approaches could be combined into an even more effective approach. We also delve into the reliability issue, specifically in decreasing the production of nonsense distractors, which has implications for teachers’ trust in the distractor generation tools. To guide our investigation, we formulate the following research questions (RQs):

1. **RQ1:** In comparison to ranking-based models, does ChatGPT generate high-quality distractors for educational MCQs?
2. **RQ2:** To what extent can we rely on ChatGPT-generated distractors, and how can we measure their trustworthiness?
3. **RQ3:** Is it possible to enhance the capability of distractor generation by combining ranking-based models with LLMs?

To answer the RQs, we designed ChatGPT prompting strategies and we solicited feedback from human experts, i.e., teachers, to evaluate the quality of generated distractors. We also compared the different strategies in terms of the reliability of generating less nonsensical distractors. In general, we found ChatGPT-driven solutions produced high-quality distractors compared to ranking-based models. They are also more reliable than the ranking-based model as they produce significantly less number of nonsense distractors. We also combined the rank-based approach with ChatGPT, through the automatic composition of an example-based prompt from the output of the rank-based model. We found that this leads to a more reliable and effective generation of distractors. The contribution of this paper can be summarized as follows:

- We proposed a strategy to guide LLMs, specifically ChatGPT, to generate effective distractors for MCQs across various subjects by prompting the model with question items automatically retrieved from existing question banks.
- We performed a user study with teachers to evaluate the quality of distractors proposed by our strategy.
- The evaluation of our approach unveils its dual capability to generate valuable distractors while simultaneously minimizing the occurrence of nonsensical options.

The remainder of the paper is organized as follows: Section 2 describes the relevant work in distractor generation and LLM prompting strategies. Section 3 explains the details of the baselines and the proposed method, while Section 4 introduces the test dataset and the evaluation setup of the user study with teachers. In Section 5.2, we report the results and provide some insights. Finally, in Section 6, we present the conclusion by summarizing the key findings and implications of our study.

## 2 Related Work

Since we briefly covered the broad application of LLMs in education in the introduction, in this section we only focus on describing prior works on distractor generation (Section 2.1) and discussing LLMs’ prompting strategies (Section 2.2) and their relevance to our work.

### 2.1 Distractor Generation

We focus on generating incorrect options (i.e., distractors) for multiple-choice questions (MCQs), which is a time-consuming task that impacts MCQ quality and has been extensively researched. Broadly speaking, the main methods for generating distractors can be categorized into retrieval-based and generation-based techniques.

*Retrieval-based methods* generate distractors by selecting the most similar alternative answers in existing knowledge bases or question item corpora. To approximate the similarity between distractors and the answer key (and question stem), several approaches are used based on (i) embedding space proximity [3, 12, 13], (ii) similarity in lexical databases such as WordNet [20], which is of particular importance in language and vocabulary learning [21, 26], and (iii) the semantic distance within domain-specific ontologies, which is critical in factoid-type questions [1, 8, 16, 25]. This ultimately leads to the selection of candidate distractors based on a ranking strategy [18].

*Generation-based methods* make use of deep learning models to directly generate distractors. Pioneering research [9, 36, 37] demonstrated the feasibility of using sequence-to-sequence models to generate distractors, while more recently, solutions based on BERT [7, 14] or T5 [30] have been explored. Rather than directly (auto-regressively) generate a distractor, the technique of back translation has shown to be relatively effective (beating a BERT-based baseline) for fill-in-the-blank language assessment tests [24].

In this work, we investigate the potential of ChatGPT<sup>2</sup>, a large and autoregressive language model, in creating distractors. We aim to combine retrieval-based and generative-based approaches by (i) automatically retrieving similar question items from pre-existing question banks to compose an example prompt and (ii) using this example prompt to guide ChatGPT to generate relevant distractors.

### 2.2 Prompting strategies

Recent instruction-based large language models (LLMs) have been a game-changer for various tasks, showing remarkable performance without any task-specific training (e.g., through finetuning) of the LLM [4, 27]. A specific task is solved through phrasing an instruction (zero-shot), possibly including a few input/output examples (few-shot) for the task at hand, as the so-called prompt

<sup>2</sup> <https://chat.openai.com/>

that serves as input to the LLM. The few-shot setting, including some examples, is commonly referred to as in-context learning (ICL). Another prompting strategy, chain-of-thought, induces language models to generate intermediate steps before predicting the final response [34].

In this paper, we introduce a variant of ICL wherein the examples presented to the LLM are determined dynamically, based on the test example (i.e., the question to generate distractors for, in our case).

### 3 Methods

We now describe our finetuned T5-based model (Section 3.1), and out-of-the-box ChatGPT-based solutions in a zero-shot setting (Section 3.2), as well as using in-context learning (Section 3.3).

#### 3.1 T5-based Distractor Generation

We fine-tuned a multilingual T5 (mT5) model [35] to generate distractors. To this end, we use a private dataset (i.e., the Televic dataset from [3]) of 62K multiple-choice question items in the form of triplets comprising a question, answer and distractors. These question items are diverse in terms of language, domain, subject and question type. On average, a question item has more than 2 distractors and contains exactly one answer. Additionally, the distractors in the dataset are not limited to single-word distractors.

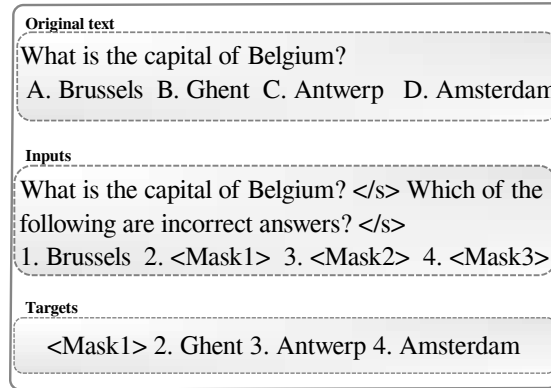
Following the unsupervised pre-training objective used in the mT5 model, we rearranged our fine-tuning data into input and output sequences as illustrated in Fig. 1. Our mT5 model’s input sequence is constructed by copying the question stem and answer from the original question item and inserting the sentence “Which of the following are incorrect answers” (or its translation depending on the language of the question item) between them. Furthermore, we masked each distractor (i.e., distractors could be multi-word spans) in the question item using a sentinel token<sup>3</sup> and separated them by increasing item numbers. The target sequence corresponds to all the dropped-out distractors and the objective is to predict the distractors.

The fine-tuning configuration that we have devised is intended to simplify the generation of multiple distractors. Specifically, all the necessary distractors for each question are generated as a list separated by numbers in a single decoding step.

#### 3.2 Zero-shot ChatGPT

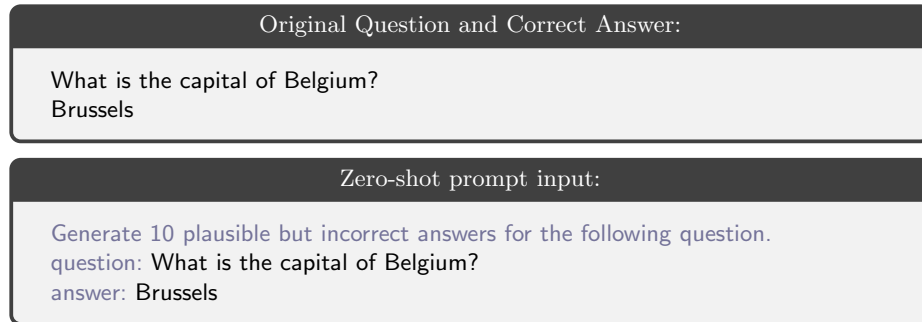
To use ChatGPT in a zero-shot setting (Zero-ChatGPT), we construct a prompt that concatenates a fixed instruction sentence and the test example, as shown

<sup>3</sup> Each sentinel token is assigned a token ID that is unique to the sequence. The sentinel IDs are special tokens added to the model’s vocabulary and do not correspond to any wordpiece



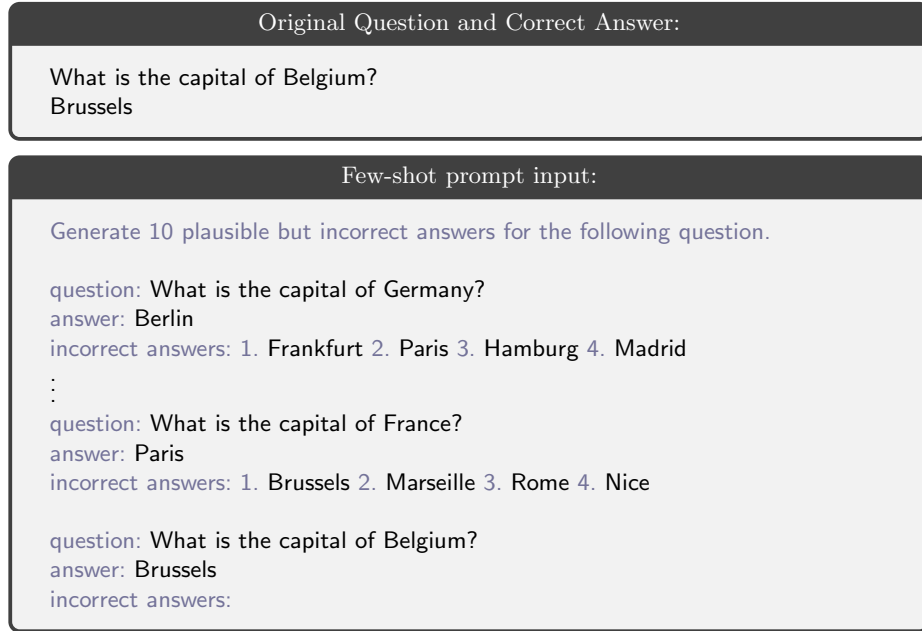
**Fig. 1.** Schematic of our fine-tuning procedure. The input sequence is constructed by copying the question and the answer from the original text and adding the template sentence “Which of the following are incorrect answers”. Each distractor is masked with a unique sentinel token (shown as  $\langle \text{Mask}_x \rangle$ ). The output sequence then consists of the dropped-out distractors. Note that a single sentinel token replaces all consecutive spans of dropped-out tokens, and the template sentence is translated into the language of the question item (i.e., Dutch or French).

in Fig. 2. Note that each time a new query is made to ChatGPT, we clear conversations to avoid the influence of previous samples through independent API calls. We use a Python ChatGPT wrapper<sup>4</sup> to call the ChatGPT API automatically.



**Fig. 2.** Example of a question with its correct answer and how we turn that into a zero-shot prompt. Note that we translate the fixed template parts for questions in languages other than English.

<sup>4</sup> Note that all the calls to the API were made between 06/04/2023 and 11/04/2023. Link to wrapper: <https://github.com/mmabrouk/chatgpt-wrapper>



**Fig. 3.** Schematic of our demonstration-based prompt construction. The top- $k$  example demonstrations are automatically retrieved from the Televic question pool, and concatenated with the instruction and test instance. This prompt is used as a query to ChatGPT for generating distractors. Note that the *fixed template* parts are translated into the language of the test question item (i.e., Dutch or French).

### 3.3 Demonstration-based ChatGPT

Finally, we evaluate ChatGPT in a few-shot setting by probing it with smartly chosen demonstrations (Dynamic-Demo-ChatGPT). We propose to retrieve the most relevant question items from the Televic dataset (see Section 3.1) and use them as demonstrations for a given test instance. We accomplish this by leveraging the question similarity (Q-SIM) model proposed by [3] to automatically select the top similar question items for the given test instance. The Q-SIM model is a BERT-based ranking model that returns a ranked list of question items according to their similarity to a given test question. Figure 3 illustrates how we combine the original question (to generate distractors for) with the retrieved examples into a prompt to ChatGPT.

## 4 Experiments

### 4.1 Test Dataset

To quantitatively evaluate our distractor generating models introduced in Section 3, we use the Wezooz test data introduced by [3], which comprises 300

**Table 1.** Inter-annotation agreement of experts, measured by the Jaccard similarity coefficient.

Subjects	True	Good	Poor	Nonsense	Overall
English	50.0	37.6	8.9	40.0	49.4
Geography	33.3	75.7	34.1	35.0	74.0

multiple-choice questions (MCQs) designed for language and factual knowledge learning and is aimed at secondary school students and teachers. It includes French and English questions for language learning purposes, while Natural sciences, Geography, History and Biology constitute the factoid questions. Each subject has 50 MCQs. Note that the data distribution of the factoid questions is different from the Televic dataset (see Section 3.1 for details), which we use to (i) fine-tune our mT5 model, and (ii) retrieve similar examples in our demonstration-based ChatGPT model. However, the language learning questions are drawn from the same distribution, in a similar design setup as [3].

## 4.2 Human Expert Quality Assessment

We also investigated our models’ output quality using human assessors, by collecting feedback from teachers. For each of the 300 questions in the aforementioned WeZooz test set, we generated 10 distractors with each of our 3 models. The teachers were then presented with a randomized list of all 30 generated distractors for each question. They were explicitly instructed to rate each distractor independent of the other distractors in the list, based on how much they thought it would help them if they were given the task of preparing distractors for that specific question. We used the four-level annotation scheme proposed by [3] to assign quality labels to each distractor: (1) **True Answer**: the distractor partially or completely overlaps with the answer key. (2) **Good distractor**: the distractor is viable and could be used in an MCQ as is. (3) **Poor distractor**: the distractor is on topic but could easily be ruled out by students. (4) **Nonsense distractor**: distractor is completely out of context.

## 5 Results and Discussion

In this section, we provide evidence of the effectiveness and reliability of our approach by reporting the experimental results and discussing the insights obtained. In Section 5.1, we explain the annotation agreement among the teachers, followed by the evaluation results in Section 5.2.

### 5.1 Inter-annotator agreement

Following the annotation scheme introduced in Section 4.2, a total of 12,860 ratings for distractor quality were collected from the annotation by teachers (see



**Table 2.** Expert evaluation of distractors (%). GDR: good distractor rate, NDR: non-sense distractor rate;  $\uparrow$ : higher is better,  $\downarrow$ : lower is better; evaluation on WeZooz test set. The markers  $\star$  and  $\ddagger$  respectively denote the one-tailed significance levels of the bootstrap-based  $p$ -value, i.e.,  $p < 0.1$  and  $p < 0.01$  with respect to the best model Dynamic-Demo-ChatGPT in each column.

Models	Language learning		Factoid learning	
	GDR@10 $\uparrow$	NDR@10 $\downarrow$	GDR@10 $\uparrow$	NDR@10 $\downarrow$
DQ-SIM [3]	27.9 $\ddagger$	44.6 $\ddagger$	28.9 $\ddagger$	50.1 $\ddagger$
mT5	24.5 $\ddagger$	42.3 $\ddagger$	27.8 $\ddagger$	36.6 $\ddagger$
Zero-ChatGPT	30.2 $\ddagger$	34.6 $\ddagger$	57.6 $\star$	17.5 $\star$
Dynamic-Demo-ChatGPT	<b>46.7</b>	<b>15.5</b>	<b>58.8</b>	<b>16.4</b>

Table 4 in Appendix A for details of rating statistics). These ratings come from 10 distractors generated by each of the models (i.e., all presented simultaneously to teachers as randomly shuffled list). In total, 10 teachers participated in our quality assessment study.

We adopt two strategies to determine the level of agreement between annotators. First, we ask teachers to rate the same set of distractors using the four-level annotation scale. We selected the subjects English, from language category, and Geography, from factoids, for annotations by at least two teachers. Table 1 shows the inter-annotator agreement of teachers using the Jaccard similarity coefficient. The Jaccard similarity measures the similarity between two sets of data by calculating what fraction of the union of those datasets is covered by their intersection. In our case, it is calculated as the number of times the teachers agreed on a distractor quality label (i.e., one of the four labels), divided by the total number of distractors that were annotated (by either annotator) with that label. In general, we note a higher agreement on what is considered a good distractor compared to the other distractor categories. Moreover, the overall agreement between the Geography teachers is higher than the English teachers.

Second, we employed the widely utilized Cohen’s kappa coefficient [19]. Our analysis substantiates the previously mentioned observation that annotators have a greater consensus when evaluating factoid questions compared to language-related queries as [3]. Specifically, among English teachers, the calculated Cohen’s kappa value stands at 28.9, signifying a “fair agreement” level. Similarly, Geography teachers exhibit a higher level of agreement with a Cohen’s kappa value of 52, indicating a level of agreement categorized as “moderate.”

## 5.2 Evaluation of models

Table 2 shows the expert evaluation of distractors in terms of *good distractor rate* (GDR@10), and *nonsense distractor rate* (NDR@10). GDR@10 is calculated as the percentage of distractors that were rated ‘good’ among the proposed 10 distractor for each model. Similarly, NDR@10 is calculated as the percentage of

distractors that were rated ‘nonsense’ among the 10 candidate distractors proposed by each model. We are interested in reporting the NDR metric because it could be used as a measure of the reliability of educational models, as a high occurrence of nonsense distractors may undermine users’ trust in the model. The reported metrics are averages of all the subjects in each category (i.e., French and English for language learning, and Biology, Natural Sciences, History and Geography for factoids). In the table, the upward arrow ( $\uparrow$ ) indicates larger values are desired, while the downward arrow ( $\downarrow$ ) indicates smaller values are preferred.

In general, the ChatGPT-based solutions (i.e., Zero-ChatGPT, and Dynamic-Demo-ChatGPT) were rated better in proposing plausible distractors than the baselines. They also produced fewer nonsense distractors. Particularly, the Dynamic-Demo-ChatGPT outperformed all the other models. On average, approximately 5 of its 10 proposed distractors were rated high-quality distractors and only 1.5 distractors were rated nonsense. Moreover, on average 8.5 distractors were generally found to be on-topic (i.e., distractors rated as either good or poor distractors) for our best model Dynamic-Demo-ChatGPT.

All the models are better at generating effective distractors for factoids than for language questions as shown by the higher GDR@10 results for factoids than languages. We hypothesize this is because, for factoid questions, our models are mainly tasked with generating accurately composed distractors that are contextually incorrect. In contrast, when faced with language questions, the intended distractors may possess ungrammatical attributes, posing a challenge for our models to generate text that is intentionally ungrammatical.

Our purely generative local mT5 model does not improve the DQ-SIM model (i.e., previous state-of-the-art model on the test set) at proposing good distractors (i.e., GDR@10 of 24.5 vs. 27.9 and 28.9 vs. 27.8). However, it is a more reliable model as it produces fewer nonsense distractors as illustrated by its lower NDR@10 values of 42.3 and 36.6 for languages and factoids, respectively, in contrast to the corresponding values of 44.6 and 50.1 for the DQ-SIM model. The relatively high number of nonsense distractors in DQ-SIM is partly attributed to its inherent limitation of only ranking pre-existing distractors according to their relevance to a given question, thereby lacking the ability to generate brand-new distractors.

In addition, in order to ensure the validity of the differences between the models, we carry out a bootstrap significance analysis [32] by sampling with replacement the annotation results DQ-SIM, mT5, Zero-ChatGPT, and Dynamic-Demo-ChatGPT models 1000 times. The resulting one-tailed significance levels ( $p$  values) are indicated in Table 2 by markers  $\star$  and  $\ddagger$  which respectively denote  $p < 0.1$  and  $p < 0.01$  with respect to our best model Dynamic-Demo-ChatGPT in each column.

***Effect of dynamically retrieved in-context examples*** We replace the dynamically retrieved examples with randomly selected language in-context examples from the Televic question bank, and we keep this selection constant (i.e., Static-Demo-ChatGPT) to generate distractors. Similar to the other models, we generated 10 distractors using the Static-Demo-ChatGPT model and asked teach-

**Table 3.** Effect of using dynamically retrieved in-context examples: **Dynamic-Demo-ChatGPT** vs. **Static-Demo-ChatGPT** that uses static in-context examples for language learning. The markers ‡ denotes the one-tailed significance level of the bootstrap-based  $p$ -value, i.e.,  $p < 0.01$  with respect to **Dynamic-Demo-ChatGPT**

Models	GDR@10 ↑	NDR@10 ↓
Static-Demo-ChatGPT	43.3‡	16.2
Dynamic-Demo-ChatGPT	<b>46.7</b>	15.5

ers to annotate the quality of the distractors. We focused on the language learning category as it showed a huge performance improvement when transitioning from **Zero-ChatGPT** to **Dynamic-Demo-ChatGPT**.

We observe that the **Dynamic-Demo-ChatGPT** model significantly outperforms the **Static-Demo-ChatGPT** model in generating high-quality distractors as indicated by the GDR@10 metric in Table 3. However, the difference in generating less nonsense distractor (i.e., NDR@10) is not significant. See Table 5 in Appendix A for an example of generated distractors using the approaches.

### 5.3 Discussion of Research questions

To answer **RQ1**, we compare the ChatGPT-based solutions (i.e., **Zero-ChatGPT**, **Static-Demo-ChatGPT** and **Dynamic-Demo-ChatGPT**) with the previous state-of-the-art ranking-based model, **DQ-SIM** in generating distractors. All the ChatGPT-based distractor generation strategies significantly outperform the **DQ-SIM**.

To address **RQ2**, we employ the NDR@10 metric as a proxy to measure the trustworthiness of models. Our best model produces an average of only 16% nonsense distractors, which is a remarkable improvement compared to the previously reported state-of-the-art performance of 50% NDR@10. This significant reduction of nonsense distractors can be expected to inspire more trust in the approach by teachers.

To answer **RQ3**, we compare **Dynamic-Demo-ChatGPT**, which combines a local ranking model with ChatGPT, against **Zero-ChatGPT** and **Static-Demo-ChatGPT**. As shown in Table 2 and Table 3, combining local models with ChatGPT leads to a better quality distractor generation, highlighting the effectiveness of this combined approach.

## 6 Conclusion

This research paper introduced and evaluated a novel strategy designed to guide LLMs, such as ChatGPT, in generating reliable and effective distractors for the creation of MCQs in educational contexts. Our proposed approach, **Dynamic-Demo-ChatGPT** model combines a rank-based approach with ChatGPT. This involves the dynamic retrieval of relevant question items through the ranker

that are then presented as in-context examples to ChatGPT for generating distractors. Our results indicated that the **Dynamic-Demo-ChatGPT** showed a considerably reduced production of nonsense distractors (i.e., only 16% rated as nonsense) compared to **Zero-ChatGPT** (i.e., out-of-the-box ChatGPT), which we consider a useful asset in terms of trust in the model by teachers. Moreover, on average, 5 out of the 10 distractors suggested by our approach were rated as high-quality by teachers, to be readily used.

For future work, we aim to investigate designing a fine-grained evaluation setup for distractors that takes into account various factors such as the level of the student, the difficulty of the questions etc. There is also a potential to explore alternative prompting strategies for LLMs, when generating distractors. For example, the utilization of self-correcting mechanism [33], which involves revising the initial output of an LLM by evaluating certain aspects of the text, could be explored in the context of distractor generation.

## References

1. Alsubait, T., Parsia, B., Sattler, U.: Generating multiple questions from ontologies: How far can we go? In: Proceedings from the First International Workshop on Educational Knowledge Management (EKM 2014), Linköping, November 24, 2014. pp. 19–30. No. 104, Linköping University Electronic Press (2014)
2. Bitew, S.K., Deleu, J., Dogruöz, A.S., Develder, C., Demeester, T.: Learning from partially annotated data: Example-aware creation of gap-filling exercises for language learning. In: Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023). pp. 598–609. Association for Computational Linguistics, Toronto, Canada (Jul 2023), <https://aclanthology.org/2023.bea-1.51>
3. Bitew, S.K., Hadifar, A., Sterckx, L., Deleu, J., Develder, C., Demeester, T.: Learning to reuse distractors to support multiple choice question generation in education. *IEEE Transactions on Learning Technologies* (2022). <https://doi.org/10.1109/TLT.2022.3226523>
4. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. *Advances in neural information processing systems* **33**, 1877–1901 (2020)
5. Cavalcanti, A.P., Barbosa, A., Carvalho, R., Freitas, F., Tsai, Y.S., Gašević, D., Mello, R.F.: Automatic feedback in online learning environments: A systematic literature review. *Computers and Education: Artificial Intelligence* **2**, 100027 (2021)
6. Choi, J.H., Hickman, K.E., Monahan, A., Schwarcz, D.: Chatgpt goes to law school. Available at SSRN (2023)
7. Chung, H.L., Chan, Y.H., Fan, Y.C.: A BERT-based distractor generation scheme with multi-tasking and negative answer training strategies. In: Findings of the Association for Computational Linguistics: EMNLP 2020. pp. 4390–4400. Association for Computational Linguistics, Online (Nov 2020). <https://doi.org/10.18653/v1/2020.findings-emnlp.393>
8. Faizan, A., Lohmann, S.: Automatic generation of multiple choice questions from slide content using linked data. In: Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics. pp. 1–8 (2018)

9. Gao, Y., Bing, L., Li, P., King, I., Lyu, M.R.: Generating distractors for reading comprehension questions from real examinations. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 6423–6430 (2019)
10. Gierl, M.J., Bulut, O., Guo, Q., Zhang, X.: Developing, analyzing, and using distractors for multiple-choice tests in education: A comprehensive review. *Review of Educational Research* **87**(6), 1082–1116 (2017)
11. Gilson, A., Safranek, C.W., Huang, T., Socrates, V., Chi, L., Taylor, R.A., Char-tash, D., et al.: How does chatgpt perform on the united states medical licensing examination? the implications of large language models for medical education and knowledge assessment. *JMIR Medical Education* **9**(1), e45312 (2023)
12. Guo, Q., Kulkarni, C., Kittur, A., Bigham, J.P., Brunskill, E.: Questimator: Generating knowledge assessments for arbitrary topics. In: IJCAI-16: Proceedings of the AAAI Twenty-Fifth International Joint Conference on Artificial Intelligence (2016)
13. Jiang, S., Lee, J.S.: Distractor generation for chinese fill-in-the-blank items. In: Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications. pp. 143–148 (2017)
14. Kalpakchi, D., Boye, J.: BERT-based distractor generation for Swedish reading comprehension questions using a small-scale dataset. In: Proceedings of the 14th International Conference on Natural Language Generation. pp. 387–403. Association for Computational Linguistics, Aberdeen, Scotland, UK (Aug 2021), <https://aclanthology.org/2021.inlg-1.43>
15. Kurdi, G., Leo, J., Parsia, B., Sattler, U., Al-Emari, S.: A systematic review of automatic question generation for educational purposes. *International Journal of Artificial Intelligence in Education* **30**(1), 121–204 (2020)
16. Leo, J., Kurdi, G., Matentzoglou, N., Parsia, B., Sattler, U., Forge, S., Donato, G., Dowling, W.: Ontology-based generation of medical, multi-term mcqs. *International Journal of Artificial Intelligence in Education* **29**(2), 145–188 (2019)
17. Li, Y., Sha, L., Yan, L., Lin, J., Raković, M., Galbraith, K., Lyons, K., Gašević, D., Chen, G.: Can large language models write reflectively. *Computers and Education: Artificial Intelligence* **4**, 100140 (2023)
18. Liang, C., Yang, X., Dave, N., Wham, D., Pursel, B., Giles, C.L.: Distractor generation for multiple choice questions using learning to rank. In: Proceedings of the thirteenth workshop on innovative use of NLP for building educational applications. pp. 284–290 (2018)
19. McHugh, M.L.: Interrater reliability: the kappa statistic. *Biochemia medica* **22**(3), 276–282 (2012)
20. Miller, G.A.: Wordnet: a lexical database for english. *Communications of the ACM* **38**(11), 39–41 (1995)
21. Mitkov, R., Varga, A., Rello, L., et al.: Semantic similarity of distractors in multiple-choice tests: extrinsic evaluation. In: Proceedings of the workshop on geometrical models of natural language semantics. pp. 49–56 (2009)
22. OpenAI: Gpt-4 technical report (2023)
23. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al.: Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems* **35**, 27730–27744 (2022)
24. Panda, S., Palma Gomez, F., Flor, M., Rozovskaya, A.: Automatic generation of distractors for fill-in-the-blank exercises with round-trip neural machine translation. In: Proceedings of the 60th Annual Meeting of the Asso-

- ciation for Computational Linguistics: Student Research Workshop. pp. 391–401. Association for Computational Linguistics, Dublin, Ireland (May 2022). <https://doi.org/10.18653/v1/2022.acl-srw.31>
25. Papasalouros, A., Kanaris, K., Kotis, K.: Automatic generation of multiple choice questions from domain ontologies. *e-Learning* **1**, 427–434 (2008)
  26. Pino, J., Heilman, M., Eskenazi, M.: A selection strategy to improve cloze question quality. In: *Proceedings of the Workshop on Intelligent Tutoring Systems for Ill-Defined Domains. 9th International Conference on Intelligent Tutoring Systems*, Montreal, Canada. pp. 22–32. Citeseer (2008)
  27. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. *OpenAI blog* **1**(8), 9 (2019)
  28. Ramesh, D., Sanampudi, S.K.: An automated essay scoring systems: a systematic literature review. *Artificial Intelligence Review* **55**(3), 2495–2527 (2022)
  29. Ramsden, P.: *Learning to teach in higher education*. Routledge (2003)
  30. Rodriguez-Torrealba, R., Garcia-Lopez, E., Garcia-Cabot, A.: End-to-end generation of multiple-choice questions using text-to-text transfer transformer models. *Expert Systems with Applications* **208**, 118258 (2022)
  31. Roediger III, H.L., Karpicke, J.D.: Test-enhanced learning: Taking memory tests improves long-term retention. *Psychological science* **17**(3), 249–255 (2006)
  32. Sakai, T.: Evaluating information retrieval metrics based on bootstrap hypothesis tests. *IPSJ Digital Courier* **3**, 625–642 (2007)
  33. Wang, R., Wang, H., Mi, F., Chen, Y., Xu, R., Wong, K.F.: Self-critique prompting with large language models for inductive instructions. *arXiv preprint arXiv:2305.13733* (2023)
  34. Wei, J., Wang, X., Schuurmans, D., Bosma, M., brian ichter, Xia, F., Chi, E.H., Le, Q.V., Zhou, D.: Chain of thought prompting elicits reasoning in large language models. In: *Advances in Neural Information Processing Systems* (2022), [https://openreview.net/forum?id=\\_VjQlMeSB\\_J](https://openreview.net/forum?id=_VjQlMeSB_J)
  35. Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., Raffel, C.: mT5: A massively multilingual pre-trained text-to-text transformer. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. pp. 483–498. Association for Computational Linguistics, Online (Jun 2021). <https://doi.org/10.18653/v1/2021.naacl-main.41>
  36. Yeung, C.Y., Lee, J.S., Tsou, B.K.: Difficulty-aware distractor generation for gap-fill items. In: *Proceedings of the The 17th Annual Workshop of the Australasian Language Technology Association*. pp. 159–164 (2019)
  37. Zhou, X., Luo, S., Wu, Y.: Co-attention hierarchical network: Generating coherent long distractors for reading comprehension. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 34, pp. 9725–9732 (2020)

## A User Study Details

This section contains the user study details. Table 4 describes the data gathered from the annotations provided by teachers. Every subject contains 50 questions, except English which has 48 questions. We collected 12,860 annotations for the proposed candidate distractors (i.e., 10 distractors by each of the three models). A total of 10 teachers participated in the study. English (i.e., from languages) and Geography (i.e., from factoids) were annotated twice by two different teachers to

calculate inter-annotator agreement. Additionally, to study the effect of dynamic retrieval of in-context examples, we asked 1 English and 1 French teacher to annotate the distractor predictions from the **Static-Demo-ChatGPT** model. The second column (i.e., *Item count*), shows the number of question items for each subject in the Wezooz dataset. Alongside, the *distractors count* column provides two distinct values: the gold truth distractors count within the dataset, and the count of unique distractors generated by our models. It is important to note that different models may produce identical distractors for a given question, resulting in varying numbers of newly generated distractors across the different subjects.

**Table 4.** Ratings Data Description

Subjects	Item count	Distractors count		Ratings count	No of Raters
		Gold	Generated		
English	48	130	1324	3360	3
French	50	92	1396	2000	2
Geography	50	145	1248	3000	2
History	50	130	1354	1500	1
Biology	50	88	1391	1500	1
Nat. Sciences	50	100	1449	1500	1
Total	298	685	8162	12860	10

## B Example Generated Distractors

Table 5 shows a cherry-picked example of our ChatGPT-based models.

**Table 5.** Some generated examples from Zero-ChatGPT, Dynamic-Demo-ChatGPT, Static-Demo-ChatGPT models for English. High-quality distractors are shown in **bold-face**, while on-topic and nonsense distractors are *italicized* and underlined, respectively. We only show 2 in-context examples for the Static-Demo-ChatGPT and Dynamic-Demo-ChatGPT models as part of the prompt but in practice, we use 5 of such examples.

Model	Example
Zero-ChatGPT	<p><b>Prompt:</b> Generate 10 plausible but incorrect answers for the following question.  question: They do n't speak English and can not make ... understood.  answer: themselves  <b>Ground:</b> [them, us, you]  <b>Prediction:</b> [1. <i>each other</i>, 2. <i>any sense</i>, 3. <u>a sentence</u>, 4. <u>appropriate gestures</u>, 5. <u>complete thoughts</u>, 6. <u>proper communication</u>, 7. <u>accurate translations</u>, 8. <u>meaningful conversation</u>, 9. <u>clear statements</u>, 10. <u>intelligible sounds</u>]</p>
Static-Demo-ChatGPT	<p><b>Prompt:</b> Generate 10 plausible but incorrect answers for the following question.  question: Why do n't we ask Jane to come over and talk ? She 's an ...  answer: alumna  incorrect answers: 1. alumnus 2. alumni 3. alumn  question: Choose the opposite of the word in bold . This company is loss-making .  answer: profit-making  incorrect answers: 1. loose 2. lucrative 3. unprofitable  ...  question: They do n't speak English and can not make ... understood.  answer: themselves  <b>Ground:</b> [them, us, you]  <b>Prediction:</b> [1. <i>themselvess</i>, 2. <b>themselves</b> 3. <b>themselves</b>, 4. <b>themselves</b>, 5. <i>themselfis</i>, 6. <i>themselfies</i>, 7. <b>themselves</b>, 8. <i>themselfselves</i>, 9. <u>themselfiness</u>, 10. <u>themselfulated</u> ]</p>
Dynamic-Demo-ChatGPT	<p><b>Prompt:</b> Generate 10 plausible but incorrect answers for the following question.  question: They should choose ... would be best suited for the job  answer: which  incorrect answers: 1. whom 2. that 3. him who  question: 36 . He does not speak English and can not make ... understood .  answer: himself  incorrect answers: 1. you 2. us 3. herself  question: The new e-ticketing system , ... .. was installed last month , is working well .  answer: which  incorrect answers: 1. what 2. him 3. where  ...  question: They do n't speak English and can not make ... understood.  answer: themselves  <b>Ground:</b> [them, us, you]  <b>Prediction:</b> [1. <b>theirself</b>, 2. <b>theirselves</b>, 3. <b>themself</b>, 4. <b>themselves</b>, 5. <b>theirs</b>, 6. <b>us</b>, 7. <u>they</u>, 8. <u>them is</u>, 9. <b>theyself</b>, 10. <b>themselves</b> ]</p>