Beyond Fine-Tuning: A Non-Parametric Approach to Distractor Synthesis for Multiple-Choice Questions

Yu-Chen Cheng
National Chung Hsing University

Yao-Chung Fan[*]
National Chung Hsing University yfan@nchu.edu.tw

Abstract

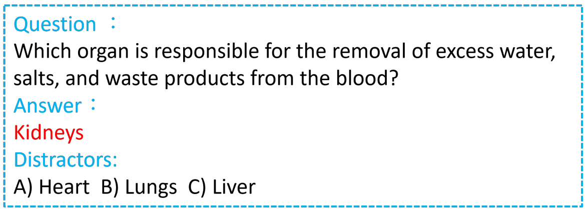
Automated distractor generation is crucial for creating effective multiple-choice questions. Traditional methods often require fine-tuning language models with domain-specific data, limiting adaptability and scope control. This paper introduces a new, non-parametric framework that uses machine reading comprehension to generate contextually relevant yet incorrect distractors from hard negative passages, without the need for fine-tuning. This approach allows rapid deployment across various domains and enables educators to tailor questions to specific content. Our framework outperformed state-of-the-art models by 8 percentage points on in-domain datasets and 75 percentage points on out-domain datasets, offering greater adaptability and controllability, making it more suitable for educational use.

Keywords: Multiple-Choice Questions, Distractor Generation, Non-parametric Framework

1 Introduction

Effective distractor selection are crucial in automated assessment systems for evaluating the depth of a learner's understanding. Creating such distractors traditionally requires expert human, making it a significant challenge in automated systems. Therefore, there has been an increasing focus on automating distractor generation through research (Liang et al., 2017, 2018; Chiang et al., 2022; Wang et al., 2023).

The common setting of multiple choice question (MCQ) is as shown in Figure 1. For automatic generation, the setting of distractor generation (DG) task is to take (1) a question stem Q and (2) the corresponding answer A as input. The goal of the output is to have a set of distractors (should be relevant but wrong with respect to Q and A).



Question ：
Which organ is responsible for the removal of excess water, salts, and waste products from the blood?
Answer ：
Kidneys
Distractors:
A) Heart B) Lungs C) Liver

Figure 1: A MCQ Test Example: the challenge to MCQ test preparation lies in wrong option (distractor) selection.

The existing DG methods, as discussed in recent studies (Wang et al., 2023; Ren and Q. Zhu, 2021; Chung et al., 2020), involve fine-tuning language models (LMs) using specialized DG datasets (such as CLOTH (Xie et al., 2017) or MCQ (Ren and Q. Zhu, 2021)), as illustrated in the left part of Figure 2. Despite the progress in current DG methodologies, there is still significant room for enhancement. First, fine-tuning approaches require domain-specific adaptation when transitioning to new fields. For instance, a DG model trained for the medical field cannot be directly applied to generate content for the scientific domain. Second, certain educational scenarios demand control over the question scope, such as restricting the generated questions and distractors to a specific range or the currently taught scope.

Addressing these limitations, we propose a non-parametric framework that begins by retrieving relevant contexts and then *extracting wrong answers* from these contexts. At the heart of our framework is the use of an *Extractive Reader*. In the context of machine reading comprehension (MRC) (Zhang et al., 2021), an

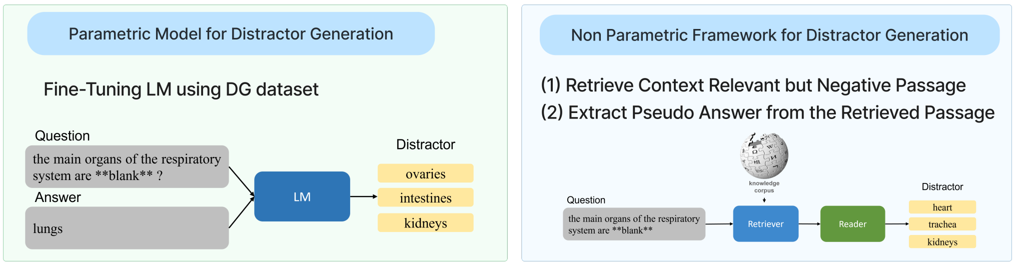

Parametric Model for Distractor Generation
Fine-Tuning LM using DG dataset
Question
the main organs of the respiratory system are **blank** ?
Answer
lungs
LM
Distractor
ovaries
intestines
kidneys
Non Parametric Framework for Distractor Generation
(1) Retrieve Context Relevant but Negative Passage
(2) Extract Pseudo Answer from the Retrieved Passage
knowledge corpus
Question
the main organs of the respiratory system are **blank**
Retriever
Reader
Distractor
heart
trachea
kidneys

Figure 2: Contextual Distractor Synthesizer: This figure contrasts our proposed non-parametric framework (the left part of the figure) for distractor generation (right) with the existing fine-tuned Language Models (LM) approach (left) (Wang et al., 2023; Ren and Q. Zhu, 2021; Chung et al., 2020). Our framework operates by first retrieving contextually relevant passages that do not contain the correct answers. It then extracts pseudo answers from these passages with a machine reading comprehension model, utilizing them as potential distractor candidates.

Extractive Reader is a type of model designed to answer questions by identifying and extracting spans of text directly from a given document. This approach assumes that the answer to any question is a span of the document's text—typically a phrase or a sentence.

We utilized an *Extractive Reader* as the *Contextual Distractor Synthesizer*. Pre-selected hard negative passages and question stems were input into the *Contextual Distractor Synthesizer*, enabling us to generate pseudo answers—plausible but incorrect distractors.

Distractor Synthesis Process: Upon receiving a relevant passage (which does not contain the correct answer) and a question stem, the Contextual Distractor Synthesizer generates pseudo answers that are contextually relevant to the question stem, using the passage's content to ensure their plausibility as distractors. These pseudo answers are incorrect by design, serving as high-quality distractors.

The use of hard negative passages in our methodology ensures a balance between relevance and incorrectness, creating distractors that are both challenging and convincing. This approach significantly increases the cognitive demands on learners, thereby improving the quality of multiple-choice assessments.

The following features distinguish our DG method:

- **No Need for Fine-Tuning:** Setting our method apart from other DG models that require extensive fine-tuning, our approach operates without the need for specific training. This attribute allows for swift adaptability across various domains, as changing the knowledge corpus is sufficient to tailor distractor generation to different subject areas.

- **Control over Question Scope:** Our framework offers the unique ability to control the thematic scope of the questions. By exchanging the data corpora, we can selectively focus on generating questions within specific topic boundaries. This aspect of our methodology resonates with the practical necessities of educational contexts, where educators often seek to limit question topics to relevant subject matter. This level of control over the question scope is a feature not readily achievable in Text2Text models(Wang et al., 2023).

2 Related Work

The landscape of distractor generation research is currently delineated into two primary frameworks, each with its distinct methodologies and advancements.

Generating and Ranking (GR) Framework: This framework employs a general-purpose knowledge base to create a candidate set of distractors, followed by a feature-rich learning-to-rank model for distractor selection. The GR architecture operates in two stages: initial generation of candidate distractors and subsequent ranking based on semantic rules and linguistic features. There are two prevalent approaches within this framework: one utilizes a knowledge base (Ren and Zhu, 2021),

	Method Type		Adaptability	Controlability	Model
	Extractive	Generative			Type
Gao et al. 2019		Y			RNN
Zhou et al. 2020		Y			RNN
Araki et al. 2016	Y		Y		Non-neural model
Welbl et al. 2017	Y				Random forests
Guo et al. 2016	Y				Word2Vec
Kumar et al. 2015	Y		Y		SVM
Liang et al. 2017		Y			GAN
Liang et al. 2018	Y		Y		neural/feature-based model
Chung et al. 2020		Y			PLM
Ren and Q. Zhu 2021		Y	Y		Knowledge-base
Peng et al. 2022		Y			PLM
Chiang et al., 2022		Y			PLM
Wang et al., 2023		Y			Text2Text
Our work	Y		Y	Y	Retriever-Reader

Table 1: An Overview of the Existing Distractor Generation Methods: Adaptability: the ability to swift over various domains. Controlability: the ability to control over generated question scope

and the other leverages a language model (Chiang et al., 2022). These approaches have marked a significant improvement over traditional rule-based methods (Liang et al., 2017, 2018), offering enhanced quality and diversity in distractors and are considered state-of-the-art in DG.

Text2Text Generation Architecture: The Text2Text generation framework formulates distractor generation as a Text2Text task, diverging from the GR approach. It involves concatenating the question stem with the correct answer and feeding this combined input into a generative language model (e.g., T5 or GPT). This architecture trains the model to generate a set of distractors in a direct and streamlined manner. Recent research adopting the Text2Text model (Wang et al., 2023) has set new benchmarks in distractor generation, demonstrating state-of-the-art performance.

Our Contribution - Contextual Distractor Synthesizer: Our work introduces a novel paradigm in the field of DG, distinct from both the GR and Text2Text architectures. At the core of our methodology is the use of a Machine Reading Comprehension (MRC) Reader, adapted as a *Contextual Distractor Synthesizer*. Unlike the GR framework, which relies on knowledge bases or LMs for initial distractor generation, our approach utilizes pre-selected hard negative passages that are contextually aligned but factually divergent from the correct answer. This method allows for the generation of pseudo answers that are inherently incorrect but contextually relevant, enhancing the cognitive challenge in assessments.

Another significant feature is that in the real world, teachers often need to create exam questions within a specified range of topics. Previous research has not considered that generated distractors need to be within a designated scope, which can cause differences between experimental results and real-world applications. Our approach is corpus-dependent, ensuring that by providing a specified corpus, the generated distractors will be 100% within the exam scope. This method more closely aligns with practical use cases.

Moreover, our approach does not require the training associated with the Text2Text models. By swapping out the knowledge corpus, our method easily adapts to various domains, offering flexibility and control over the scope of questions. This feature is particularly advantageous in educational settings where specificity and relevance are the key.

For clarity of comparison, we summarize the existing DG studies in Table 1. Our study is the only one capable of adapting to domain changes without requiring any processing or retraining, while also customizing the scope of questions for the same problem.

In Figure 5, although our performance on the in-domain dataset was comparable to that of the current SOTA DG method, we significantly excelled in the out-domain dataset.

This underscores the limited adaptability of the SOTA DG method when applied beyond its initial domain.

3 Methodology

3.1 Problem Setting and Assumptions

The methodology assumes the following inputs :

- A knowledge corpus $\mathcal{C}$, consisting of text chunks relevant to the subject matter.
- A question stem Q, representing the query to be addressed.
- An answer A, the correct response to the question stem Q.

Additionally, the methodology utilizes two functions:

- A document retriever $\mathcal{F}()$, which fetches text chunks from the corpus $\mathcal{C}$.
- An extractive document reader $\mathcal{R}(Q)$, which extracts answers from a fetched text chunk given the question stem Q.

The details of the algorithm are presented in Algorithm 1.

3.2 Retriever

The first stage of the process is the retrieval of relevant text chunks, a critical step in identifying suitable passages for distractor generation. We use the BM25 algorithm to retrieve relevant text chunks for distractor generation.

The passages containing the correct answer and synonyms of the correct answer are filtered out, and the remaining passages are referred to as hard negative passages. This ensures contextually relevant yet misleading content for effective distractor generation.

3.3 Extractive Reader as Contextual Distractor Synthesizer

At this stage, a Extractive Reader is employed as a Contextual Distractor Synthesizer (CDS). This approach deviates from traditional reading comprehension question-answering methodologies by shifting focus from extracting accurate answers to generating contextually relevant yet incorrect distractors, termed as *pseudo answer* (PA) as served as candidates for distractors. The *pseudoanswer* are generated using the following formula:

$$P_{CDS}(PA \mid C, Q) = \prod_{i=1}^{n} P(pa_i \mid pa_1, \dots, pa_{i-1}, C, Q; \theta)$$

3.4 Formation of the Pseudo Answer Set (PAS):

PAS, or the Pseudo Answer Set, is a collection of all potential distractors generated by the MRC Reader. It is formulated based on two key parameters:

- k —the number of text chunks retrieved by the Retriever
- h —the number of pseudo answers identified by the Reader for each text chunk.

Thus, PAS comprises a total of $k \times h$ pseudo answers, each representing a potential distractor candidiate derived from the hard negative passages.

Character-Level Rouge Score Evaluation: To enhance the quality and diversity of PAS, each pseudo answer undergoes a character-level Rouge score evaluation before adding into PAS by Considering each individual character as a complete word. This evaluation assesses textual similarity and ensures that the pseudo answers are distinct from one another. Specifically, If a pseudo answer candidate (pa) has a high Rouge score compared to any existing item in PAS ($PA_{\in PAS}$), indicating a significant overlap, it is discarded to prevent redundancy.

3.5 Distractor Evaluator

The culminating phase of our distractor generation process is the evaluation of the pseudo answers by the Distractor Evaluator. This component is critical for appraising the $k \times h$ pseudo answers produced by the Contextual Distractor Synthesizer, and its primary goal is to determine the most appropriate distractors for each question based on their relevance and plausibility.

Scores of Pseudo Answers: The input to the Distractor Evaluator comprises the question stem Q, the hard negative article C, the correct answer A, and the set of pseudo answers $PAS = \{PA_i\}$. The pseudo answers are

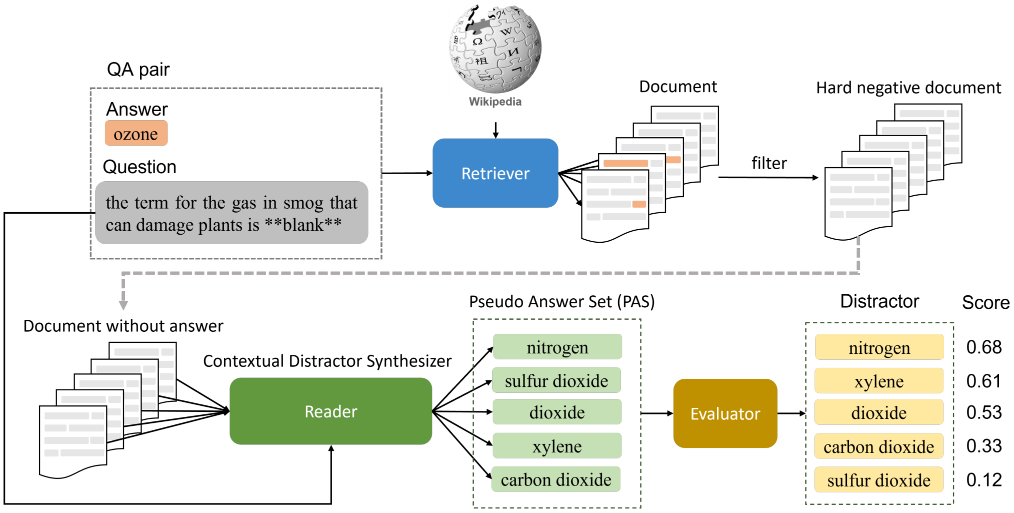


Figure 3: Overview of our approach. We can decompose our architecture into the following components: Retriever, Contextual Distractor Synthesizer, Distractor Evaluator. Initially, the retriever component searches for relevant articles and filters out those containing the correct answers, leaving only the hard negative passages. The Contextual Distractor Synthesizer component then generates pseudo answers from these hard negative passages. Finally, the evaluator component ranks the top-n distractors.

ranked based on the Confidence Score given by the following features:

- **Retriever Score S_{retr}:**

  $$S_{retr} = retr(Q, C)$$

  This score, represents the relevence between Q and C. Any value that can represent the relevance between Q and C in sparse retrieval or dense retrieval can be used.

- **Confidence Score $S_{confidence}$:**

  $$S_{confidence} = p_{CDS}(PA|Q, C; \theta)$$

  This score, reflects the confidence of the Contextual Distractor Synthesizer (CDS) in PA being a viable distractor.

Selection of Distractors: Each pseudo answer PA receives a final score based on its S_{retr} and $S_{confidence}$ scores:

$$score(PA_i) = S_{retr} \cdot S_{confidence}$$

The distractors that achieve the highest scores in this evaluation —the top-n scored items —are selected as the final distractors. This process ensures that the chosen distractors are both contextually relevant and sufficiently challenging.

4 Experiment

5 Implementation Details

In our experiments with the MCQ dataset and MEDMCQA, we utilized Wikipedia articles as the corpus for our retriever. Each article was divided into passages every five sentences, with the article title appended to the beginning of each passage. We used Pyserini to construct our BM25 retriever model, effectively identifying relevant passages from the corpus.

For the distractor synthesizer component of our framework, we selected Llama-2-7b-hf, released by Facebook Meta on Huggingface, and trained it over 2 epochs using two NVIDIA RTX 3090 GPUs with the SQuAD dataset (Rajpurkar et al., 2016). We used the question answering task to train the model. After fine-tuning the model, we can utilize relevant text chunks and questions as model inputs to generate pseudo answers. All experiments are conducted using two NVIDIA RTX GPUs.

5.1 Evaluation Metrics

We introduce the GPT-4 Distractor Effectiveness Index (GDEI) to overcome the limitations of traditional token-based metrics in evaluating distractor quality. Unlike token scores, which often miss semantic and contextual details crucial for effective distractors and are constrained by the dataset's limited ground

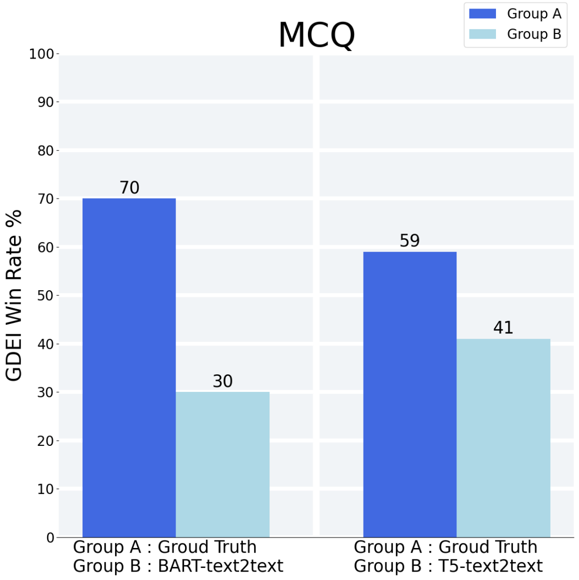


Figure 4: We aim to demonstrate the capability of the GDEI metric. Given that DG-text2text(Wang et al., 2023) which is the SOTA model employs Multiple Choice Questions (MCQ) for training, we tasked GPT-4 with comparing between the ground truth in MCQ and distractors generated by the DG-text2text. The results indicate that GPT-4 is capable of correctly differentiate instances where distractors generate by DG-text2text are lower in quality than the ground truth.

truth, GDEI utilizes GPT-4’ s advanced comprehension to assess distractor sets more holistically.

In the evaluation process, GPT-4 is given a question, the correct answer, and two sets of distractors. It then determines which set is of higher quality. To ensure fairness, the order of the distractor sets is randomly alternated. Detailed instructions are provided in Figure 10.

5.2 GDEI Validation through Comparative Analysis

Our validation contrasts GDEI scores for educator-crafted ground truth distractors against those generated by T5 and BART models. The results in Figure 4 favor ground truth distractors, showcasing GDEI's capability to discern quality reflecting human expertise in distractor design. This confirms GDEI's effectiveness as a nuanced evaluation tool for distractor generation.

5.3 Dataset

- **MCQ dataset**: MCQ dataset (Ren and Q. Zhu, 2021) is a cloze-style dataset, that includes the domains of science, vocabulary, common sense, and trivia. Each data is composed of a sentence containing " **blank** " of cloze stem, answer, and distractors.

- **MedmcQA dataset**: MedMCQA (Pal et al., 2022) MedMCQA is a vast dataset of over 194k high-quality Multiple-Choice Questions and Answers for medical entrance exam preparation. It covers 2.4k healthcare topics and 21 medical subjects from AIIMS and NEET PG exams. The questions vary in length and complexity, and each sample includes a question, correct answer(s), additional options, and a detailed solution explanation.

- **Harry Potter Novel questions**: The data from the Harry Potter novels can serve as an excellent dataset for testing Controllability of the generation method. Each book in the series contains unique plot elements not found in the others. We utilized ChatGPT to generate 100 sets of questions from the first book of the Harry Potter series, with each set comprising one question and one correct answer.

Since SOTA DG method (Wang et al., 2023) utilized the MCQ dataset (Ren and Zhu, 2021) for training, it is treated as the in-domain dataset. We chose the MCQ test set, comprising a total of 259 multiple-choice question sets, as our testing dataset.

For the adaptability experiments, we used the MCQ dataset (Ren and Zhu, 2021) as the in-domain dataset, consisting of 259 multiple-choice question sets. The MEDMCQA dataset (Pal et al., 2022) served as the out-of-domain dataset, with 100 question sets selected to test the SOTA DG model's adaptability in less related contexts.

In the controllability experiment, we generated 100 questions using ChatGPT based on the first Harry Potter book, with the scope restricted to this book only.

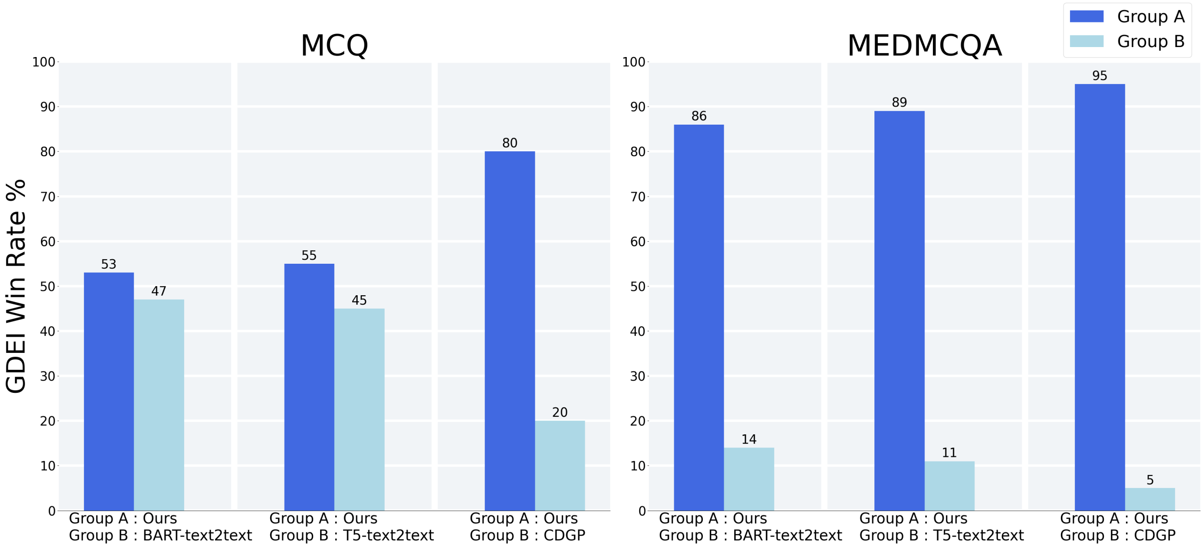


Figure 5: Experiment results of our method compared to DG-text2text(Wang et al., 2023) on both in-domain and out-domain datasets. While the margin of superiority in the in-domain dataset is modest, our method significantly outperforms the DG-text2text(Wang et al., 2023) in the out-domain dataset. This demonstrates the adaptability of our method across different domains.

5.4 Comparative Results Discussion

Our experimental analysis presents a comparative evaluation of our distractor generation method against SOTA DG model variants, namely T5 and BART, using two datasets: MCQ and MEDMCQA. The GDEI metric serves as our evaluation criterion, quantifying the effectiveness of the generated distractors.

5.4.1 Experiment : Adaptability

The overall result is shown in Figure 5.

- **Performance on MCQ Dataset:** In the MCQ experiment, our comparison targets two state-of-the-art training methods with different base models, namely T5-text2text and BART-text2text, as well as another method using the Pre-trained Language Model - CDGP (Chiang et al., 2022). In comparison with the state-of-the-art methods, our method achieved a GDEI score of 55, while the SOTA DG T5-text2text scored only 45. Our method also obtained a GDEI score of 53, whereas the SOTA DG BART-text2text scored only 47. In comparison with CDGP, our method achieved a score of 80, significantly surpassing CDGP's score of 20. This indicates a clear preference for the distractors generated by our method, suggesting that our approach produces more contextually relevant and challenging distractors.

- **Performance on MEDMCQA Dataset:** The MEDMCQA dataset further validates the superiority of our method. It attained remarkably high GDEI scores of 89 and 86, whereas the SOTA DG models scored significantly lower, with 11 for DG-T5 and 14 for DG-BART. In comparison with CDGP, our method achieved a score of 95, significantly surpassing CDGP's score of 5. The absolute contrast in the scores on this dataset emphasizes the robustness of our method in a specialized domain.

In practical educational settings, teachers across various fields need models to generate distractors. Insufficient model adaptability necessitates fine-tuning with specific distractor datasets for each field, which can be costly.

Combining results from both in-domain and out-domain datasets, our method slightly outperformed the SOTA DG model in GDEI scores within in-domain datasets but significantly surpassed it in the out-domain MEDMCQA dataset. This illustrates the limitations of traditional text-to-text methods on out-domain data, which lack adaptability. In contrast, our non-parametric architecture performs consistently across different domains without requiring fine-tuning. This stability and adaptability highlight our framework's alignment with practical educational applications, as it achieves strong performance across various domains using relevant reference articles, eliminating the need for additional manually annotated distractor datasets.

Generation Method	Distractor-wise	Question-wise
Llama 2 prompting	73%	58%
Llama 2 RAG prompting	79%	61%
Ours	100%	100%

Table 2: This table shows the probability that the generated distractors are within the specified range. The "All distractors" column indicates the probability that all 300 generated distractors are within the specified range. The "All set" column indicates the probability that all distractors for each of the 100 question sets are within the specified scope.

5.4.2 Experiment : Controllability Performance on Harry Potter Novel dataset:

Since the state-of-the-art (SOTA) distractor generation (DG) model was not trained on Harry Potter data, it struggles to generate relevant distractors for such content. To address this, we compared our extractive generation strategy with two prompting methods using Llama 2(Touvron et al., 2023) . The first method, Llama 2 prompting, involved directly inputting the question and answer to generate distractors. The second, Llama 2 RAG prompting, included a passage retrieved by BM25 with the correct answer.

We aimed to simulate realistic scenarios where teachers specify question scopes and tested our method with 100 questions from the first Harry Potter book, comparing it to the two Llama 2 strategies. Besides evaluating GDEI scores, we assessed how well the distractors matched the predefined scenario.

The results, presented in 6 and Table 2, show that although our method's GDEI scores are lower or only marginally higher compared to the state-of-the-art generation strategies, this is likely due to our method's constrained scope, which limits the generation of higher quality distractors outside this range.

Our method consistently met the specified scope conditions 100% of the time, in contrast to the Llama 2 prompting methods. With Llama 2 prompting, only 73% of distractors met the criteria of being from the first Harry Potter book, and Llama 2 RAG prompting improved this to 79%. When considering question sets, both prompting methods produced compliant distractor sets only 58% and 61% of the time, respectively.

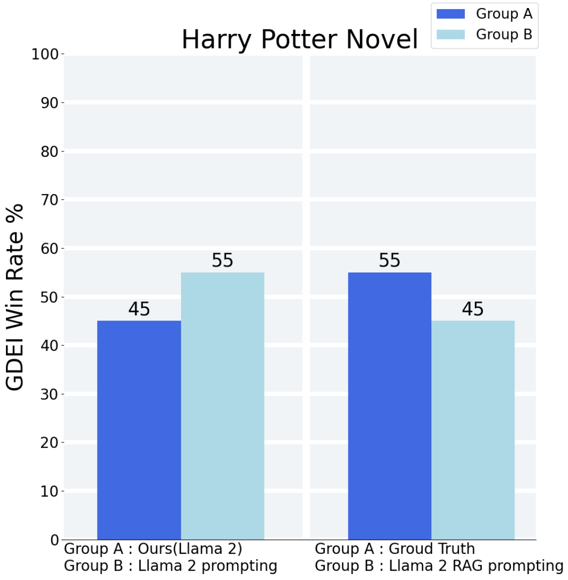


Figure 6: The comparative results of our method against two generation strategies of SOTA DG methods using llama 2 as the base model in Harry Potter Novel dataset.

These findings suggest that while large language models have parametric knowledge, they lack effective controllability for real educational scenarios. Our framework, despite lower GDEI scores, more effectively aligns with practical distractor design needs.

5.5 Case Study

5.5.1 MedMCQA case study:

In Table 3, we present a case study from the MedMCQA dataset. We can observe that in the first question, distractor set generated by our method ('perphenazine', 'penfluridol', 'chlorpromazine') is better for questions on refractory schizophrenia treatment as it includes drugs actually used for schizophrenia, requiring deeper knowledge to identify the correct answer. Distractor set generated by SOTA DG model('ibuprofen', 'tricyclics', 'Valium') includes less relevant drugs, making them less effective distractors.

In the second question, distractor set generated by our method is superior as it includes terms closely linked to protein folding, like 'Calnexin' and 'aspaate.' These are plausible distractors given their role in protein quality control. While 'Ribosome' is indirectly related, ('Xerophyte', 'Protist', 'Emb') lack relevance to protein folding, making distractor set generated by SOTA DG model less effective.

In the third question, distractor set generated by our approach not only exhibits a higher level of professionalism but also highlights the inadequacy of the SOTA distractor generation method in terms of knowledge in the out-domain scope. When encountering unfamiliar questions, the SOTA DG method may also produce redundant distractors.

5.5.2 Harry Potter Novel case study

In Table 4, we present a case study from the Harry Potter Novel experiment. In the first question example, distractor generated by our framework, is more effective because it includes items closely related to the Harry Potter universe that could be mistaken for something granting invisibility. The 'forgetfulness potion' and 'dragon egg' are magical but do not provide invisibility, making them plausible but incorrect. The 'mirror' is slightly less relevant but still within the magical realm. On the other hand, distractor generated using Llama 2 prompting, includes 'wand' and 'glasses,' which are relevant but less likely to be confused with an invisibility item, and 'crystal,' which is vague and less misleading. Distractor generated by our framework aligns better with real teacher practices as it requires students to distinguish between different magical items in the context of the story.

In the second question example, we observed that our framework produces distractors similar with those generated using the parametric knowledge of Llama 2. This demonstrates that our method, which relies on pseudo-answers extracted from the provided passages, yields results similar to those achieved with Llama 2's extensive parametric knowledge. These findings validate the effectiveness of our approach in using hard negative passages to generate distractors.

6 Conclusion

In conclusion, our DG method stands out for its unique features, notably the absence of fine-tuning requirements and the unprecedented control over question scope. Unlike other DG models, our approach operates efficiently without the need for extensive fine-tuning, allowing for swift adaptation across diverse domains by simply adjusting the knowledge corpus. The ability to selectively focus on generating questions within specific topic boundaries addresses the practical needs of educational contexts, providing educators with a level of control over question scope that conventional Text2Text models struggle to achieve. These distinctive characteristics position our DG method as a versatile and efficient tool for generating targeted and relevant questions across various subject areas.

7 Limitations

- **Assessment of Distractor Quality** Although the GDEI provides a more comprehensive evaluation of distractors, it may not capture all aspects of distractor quality, such as the potential for a distractor to reinforce common misconceptions or to be pedagogically useful.

- **Dependence on Quality of Corpus:**The quality and diversity of the generated distractors are directly tied to the richness of the knowledge corpus used. If the corpus is outdated, biased, or lacks depth, the distractors may not be as effective or may inadvertently introduce inaccuracies.

8 Appendix

A Ablation Study

A.1 Generation parameters

In figure 7, we tested three different k and h settings: k=60, h=5; k=100, h=3; and k=300, h=1. We observed that although the GDEI scores for the three settings were close when compared with SOTA-BART, the setting of k=300, h=1 significantly outperformed the other two settings when compared with SOTA-T5. We speculate that this is because we divided the corpus into small passages, each containing a limited amount of information. If h increases, it may result in meaningless distractor candidates. The optimal setting is to examine as many passages as possible and generate one distractor at a time for each passage.

A.2 Corpus chunk size

In the experiment in figure 8, we divided the corpus into chunks of varying sizes based on the number of sentences to examine the impact of chunk size on the generation results. We used the llama2-based Contextual Distractor Synthesizer with a setting of k=300 and h=1. Our data indicates that as the number of sentences per chunk decreases, the quality of the generated output improves. This outcome might be due to the extensive knowledge contained in each Wikipedia article, where each sentence potentially includes valuable words that can serve as distractor candidates. If the chunk size is too large, many potential distractors may be overlooked during generation. Conversely, smaller chunk sizes allow for a more detailed generation of high-quality distractors from each sentence within an article.

A.3 Ablation Study on different base model of the Contextual Distractor Synthesizer

We tested the impact of different base models for the Contextual Distractor Synthesizer on the generation results in figure 9. We fine-tuned Llama2-7b and Mistral-7b (Jiang et al., 2023) models for Contextual Distractor Synthesizer. Llama2-7b slightly outperformed Mistral-7b in both MCQ and MEDMCQA. We attribute this difference to the variations in the pre-training data of the lauguage model, which lead to differences in the models' s parametric knowledge. However, according to our proposed framework, as long as a Reading Comprehension model is well fine-tuned, even a smaller model can produce comparable generation results. With an adequately searchable corpus, high-quality distractors can be generated with our framework.

Algorithm 1 Distractor Generation Algorithm

```
Input: Knowledge corpus $\mathcal{C}$ consisting of text chunks, question stem $Q$, answer $A$ to $Q$
Assume:
    (1) a document retriever $\mathcal{F}()$ for fetching text chunks from $\mathcal{C}$
    (2) an extractive document reader $\mathcal{R}()$ for extracting an answer from a given text and a question
procedure GenerateDistractors($\mathcal{C}, Q, A$)
    $\kappa \leftarrow \mathcal{F}(Q)$                                ▷ Fetching Top-k text chunks
    for all $C_i \in \kappa$ do
        if $A$ is in $C_i$ then
            Remove $C_i$ from $\kappa$
        end if
    end for
    Let PAS be Pseudo Answer Set = {}
    for $C_i \in \kappa$ do
        $PseudoAnswer \leftarrow \mathcal{R}(C_i, Q)$
        Compute $S_{retr}$ for $PseudoAnswer$
        Evaluate $PseudoAnswer$ with character-level ROUGE-L score
        if $PseudoAnswer$ has low Rouge similarity with all items in PAS then
            $PAS.add(PseudoAnswer)$
        end if
    end for
    EvaluateDistractors(PAS, $Q$, $A$)
    return Top-$n$ scored items in PAS as final distractor set for $Q$ and $A$
end procedure
function EvaluateDistractors(PAS, $Q$, $A$)
    for all $PA_i \in PAS$ do
        Compute $S_{confidence}$ for $PA_i$
        $score(PA_i) \leftarrow S_{retr} \cdot S_{confidence}$                ▷ Computing final score
    end for
    Sort $PAS$ based on $score(PA_i)$ in descending order
    return Sorted $PAS$
end function
```

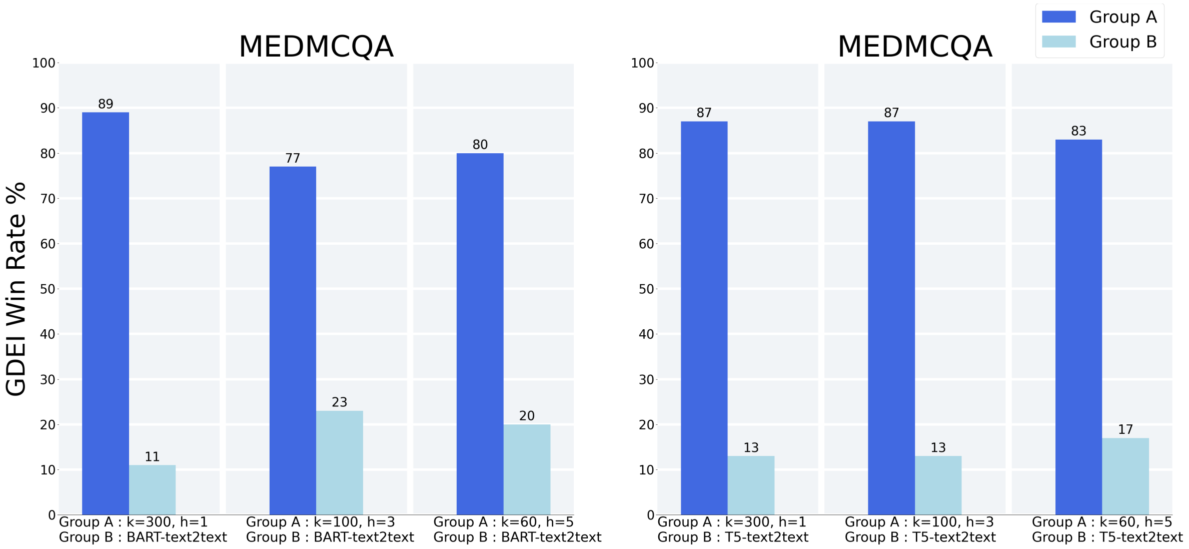


Figure 7: Experiment results of different settings of k,h parameters in comparison to DG-text2text (Wang et al., 2023) in MEDMCQA dataset.

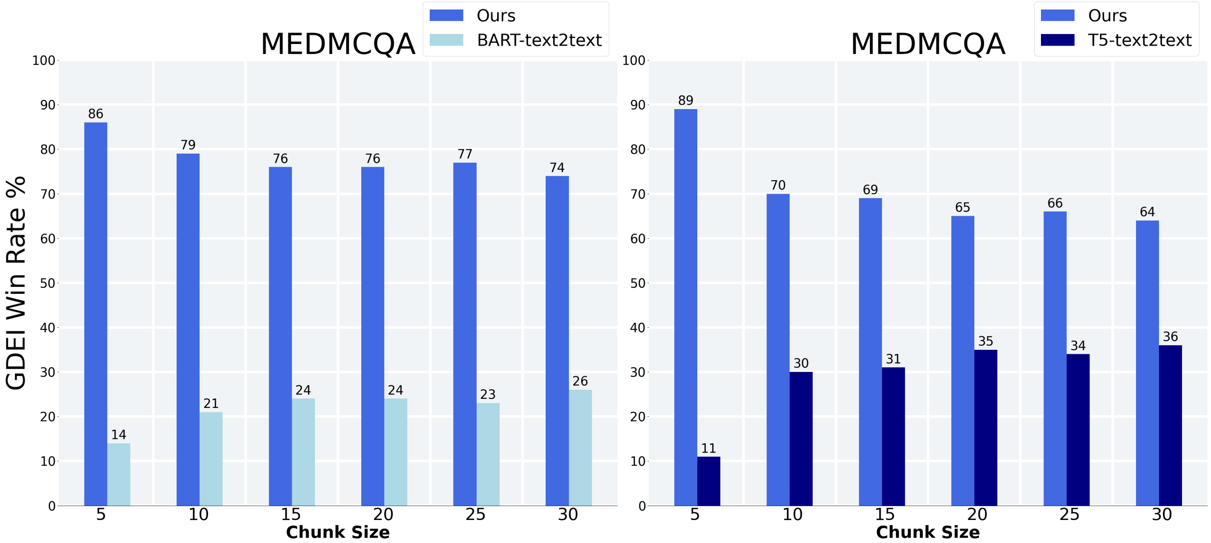


Figure 8: Different chunk sizes on the generation results in comparison to DG-text2text (Wang et al., 2023) in MEDMCQA dataset.

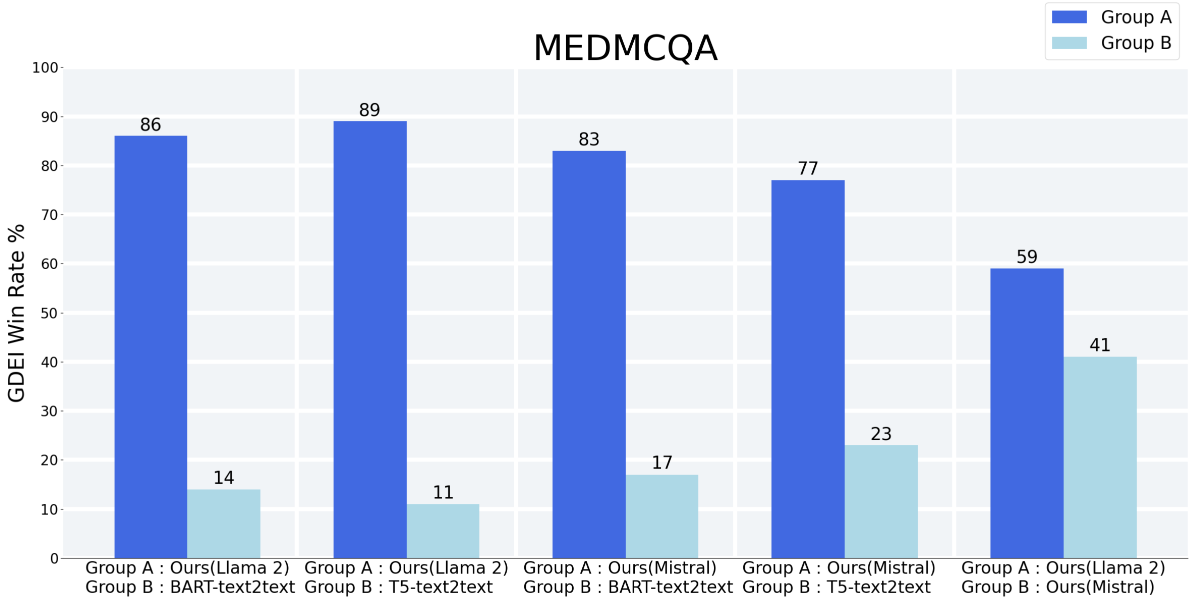


Figure 9: The performance of different base model on the Contextual Distractor Synthesizer in comparison to the SOTA DG model in MEDMCQA dataset.

```
Distractors are designed to be plausible but incorrect choices. Evaluate the quality of a distractor for the given sentence using the specified criteria.
Apply the following criteria for evaluation:
1. Exclusion from Correctness: distractors should not correctly fill the sentence.
2. Avoidance of Repetition: distractors should not be already mentioned in the sentence.
3. Relevance: distractors should be relevant to the question, not entirely unrelated or absurd choices.
4. Misleading: distractors should contain elements that may lead the test taker to select them incorrectly.
5. Alignment with Real Teacher Practices: Distractors must reflect the approach of real teachers in
crafting questions tailored for students.
Question: {question}
Answer: {answer}
Distractor set 1: { distractor set 1 }
Distractor set 2: { distractor set 2}
From a teacher's perspective, which of the following sets do you think is better: Distractor set 1 or 2.
Please output according to the following format.
reason: _(up to 100 words)
choose:set 1 or set 2 or both are same
```

Figure 10: Prompt for GPT-4 distractor evaluation for adaptability experiment.

```
You are the author of Harry Potter
Distractors are designed to be plausible but incorrect choices. Evaluate the quality of a distractor for the given sentence using the specified criteria.
Apply the following criteria for evaluation:
1. Exclusion from Correctness: distractors should not correctly fill the sentence.
2. Avoidance of Repetition: distractors should not be already mentioned in the sentence.
3. Relevance: distractors should be relevant to the question, not entirely unrelated or absurd choices.
4. Misleading: distractors should contain elements that may lead the test taker to select them incorrectly.
5. Alignment with Real Teacher Practices: Distractors must reflect the approach of real teachers in crafting questions tailored for students.
Question: {question}
Answer: {answer}
Distractor set 1: { distractor set 1 }
Distractor set 2: { distractor set 2}
From a teacher's perspective, which of the following sets do you think is better: Distractor set 1 or 2.
Please output according to the following format.
reason: _(up to 100 words)
choose:set 1 or set 2 or both are same
```

Figure 11: Prompt for GPT4 distractor evaluation for Controllability experiment.

References

Jun Araki, Dheeraj Rajagopal, Sreecharan Sankaranarayanan, Susan Holm, Yukari Yamakawa, and Teruko Mitamura. 2016. Generating questions and multiple-choice answers using semantic analysis of texts. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1125–1136.

Shang-Hsuan Chiang, Ssu-Cheng Wang, and Yao-Chung Fan. 2022. Cdgp: Automatic cloze distractor generation based on pre-trained language model. In *Findings of the Association for Computational Linguistics: EMNLP 2022*.

Ho-Lam Chung, Ying-Hong Chan, and Yao-Chung Fan. 2020. A bert-based distractor generation scheme with multi-tasking and negative answer training strategies. *arXiv preprint arXiv:2010.05384*.

Yifan Gao, Lidong Bing, Piji Li, Irwin King, and Michael R Lyu. 2019. Generating distractors for reading comprehension questions from real examinations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6423–6430.

Qi Guo, Chinmay Kulkarni, Aniket Kittur, Jeffrey P Bigham, and Emma Brunskill. 2016. Questimator: Generating knowledge assessments for arbitrary topics. In *IJCAI-16: Proceedings of the AAAI Twenty-Fifth International Joint Conference on Artificial Intelligence*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Girish Kumar, Rafael E Banchs, and Luis Fernando D' Haro. 2015. Revup: Automatic gap-fill question generation from educational texts. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 154–161.

Chen Liang, Xiao Yang, Neisarg Dave, Drew Wham, Bart Pursel, and C Lee Giles. 2018. 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*, pages 284–290.

Chen Liang, Xiao Yang, Drew Wham, Bart Pursel, Rebecca Passonneau, and C Lee Giles. 2017. Distractor generation with generative adversarial nets for automatically creating fill-in-the-blank questions. In *Proceedings of the Knowledge Capture Conference*, pages 1–4.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2022. Medmcqa: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Conference on Health, Inference, and Learning*, pages 248–260. PMLR.

Hsien-Yung Peng, Ho-Lam Chung, Ying-Hong Chan, and Yao-Chung Fan. 2022. Misleading inference generation via proximal policy optimization. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 497–509. Springer.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

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

Siyu Ren and Kenny Q Zhu. 2021. Knowledge-driven distractor generation for cloze-style multiple choice questions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4339–4347.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Hui-Juan Wang, Kai-Yu Hsieh, Han-Cheng Yu, Jui-Ching Tsou, Yu An Shih, Chen-Hua Huang, and Yao-Chung Fan. 2023. Distractor generation based on text2text language models with pseudo kullback-leibler divergence regulation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12477–12491.

question	answer	Distractor
FDA approved drug for refractory schizophrenia **blank**?	Clozapine	ours: 1.perphenazine 2.penfluridol 3.chlorpormazine
		DG-text2text (Wang et al., 2023): 1.Ibuprofen 2.Tricyclic 3.Valium
Reverse folding of proteins is carried out by **blank**?	Chaperone	ours: 1.calnexin 2.ribosome 3.aspaate
		DG-text2text (Wang et al., 2023): 1.Xerophyte 2.Protest 3.Emb
Drug of choice of benzodiazepine poisoning is **blank**?	Flumazenil	ours: 1.midazolam 2.imidazenil 3.naloxone
		DG-text2text (Wang et al., 2023): 1.Alcohol 2.Cigarettes 3.Alcohol

Table 3: Case study of our work comparing to the SOTA DG model in MEDMCQA dataset

question	answer	distractor
What magical item did Harry receive that made him invisible?	Invisible cloak	ours: 1. forgetfulness potion 2. dragon egg 3. mirror
		Llama 2 prompting: 1. crescent 2. crystal 3.wand
Who is the caretaker of Hogwarts?	Filch	ours: 1. rubeus hagrid 2. dumbledore 3. professor snape
		Llama 2 prompting: 1. hagrid 2. dumbledore 3. mcgonagall

Table 4: Case study of our work comparing to the SOTA DG model in Harry Potter Novel dataset

Johannes Welbl, Nelson F Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. *arXiv preprint arXiv:1707.06209*.

Qizhe Xie, Guokun Lai, Zihang Dai, and Eduard Hovy. 2017. Large-scale cloze test dataset created by teachers. *arXiv preprint arXiv:1711.03225*.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2021. Retrospective reader for machine reading comprehension. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14506–14514.

Xiaorui Zhou, Senlin Luo, and Yunfang Wu. 2020. Co-attention hierarchical network: Generating coherent long distractors for reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9725–9732.