Retrieval-Augmented Data Augmentation for Low-Resource Domain Tasks

Anonymous ACL submission

Abstract

Despite large successes of recent language models, they suffer from severe performance degeneration in low-resource settings with limited training data available. Many existing works tackle this problem by generating synthetic data from the training data and then training models on them, recently using Large Language Models (LLMs). However, in low-resource settings, the amount of seed data samples to use for data augmentation is very small, which makes generated samples suboptimal and less diverse. To tackle this challenge, we propose a novel method that augments training data by incorporating a wealth of examples from other datasets, along with the given training data. Specifically, we first retrieve relevant instances from other datasets, such as their input-output pairs or contexts, based on their similarities with the given seed data, and prompt LLMs to generate new samples with the contextual information within and across the original and retrieved samples. This approach can ensure that the generated data is not only relevant but also more diverse than what could be achieved using the limited seed data alone. We validate our Retrieval-Augmented Data Augmentation (RADA) framework on multiple datasets under low-resource settings of training and test-time data augmentation scenarios, on which it outperforms existing data augmentation baselines.

1 Introduction

Recent advances in language models (Brown et al., 2020; Touvron et al., 2023; OpenAI, 2023; Anil et al., 2023), which are trained on general text corpora, have achieved numerous successes across various natural language tasks. The common practice to further enhance their performances is to perform fine-tuning on task-specific datasets, which has been proven substantially effective regardless of model sizes (Gudibande et al., 2023; Lv et al., 2023). However, the efficacy of this fine-tuning is closely tied to the volume and quality of the data available for training. Meanwhile, in real-world scenarios, particularly in specific domains, there is often a scarcity of training instances. For example, at the beginning of a pandemic such as COVID-19, there are only a few limited training instances to fine-tune language models, despite an urgent need for tasks, such as question answering (Möller et al., 2020) (Figure 1, (A)). Yet, the manual annotation of additional training samples is costly and time-consuming, which may require domain experts.

To address this challenge, various approaches have been proposed to augment the training data automatically. These methods typically range from altering the texts of existing training samples (Sahin and Steedman, 2018; Wei and Zou, 2019b) to leveraging generative models to produce new instances for training based on initial seed samples (Yao et al., 2018; Anaby-Tavor et al., 2020; Lee et al., 2020). Also, many recent approaches have leveraged the capability of LLMs for data augmentation based on prompting, which eliminates the burden of performing task-specific training (Honovich et al., 2023a; Whitehouse et al., 2023; Lee et al., 2023). In particular, Chen et al. (2023a) has utilized the diverse prompting strategies to create a broader set of instances. However, in low-resource environments where only a limited number of training instances are available, generating new data from these minimal seed samples results in poor diversity and variation (See Figure 1, (B)). We note that a very recent approach attempts to overcome this by iteratively including generated samples as seed data for further data generation (Wang et al., 2023a). However, this approach is still ill-suited, which is not only constrained by the limited diversity of the initial seed data but also vulnerable to recursively diminishing the quality of subsequent augmentations due to the potential low-quality of prior augmentations.

Despite the limited seed data in low-resource settings, there is an abundance of examples and re-

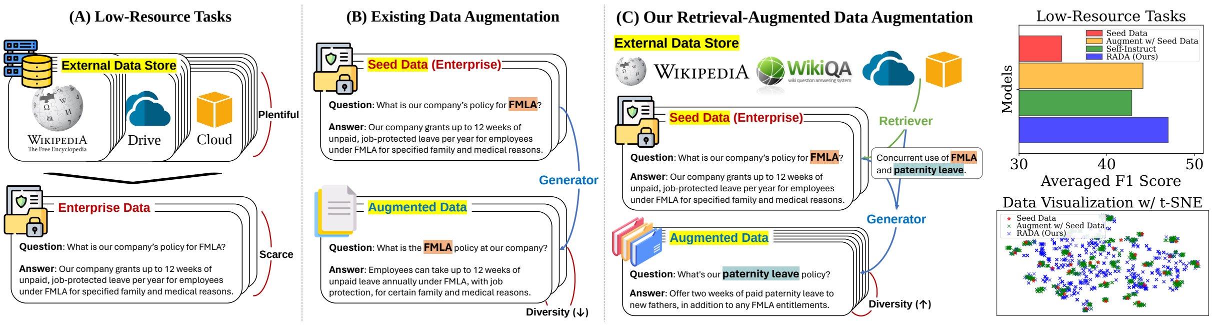

Figure 1: **(A) Low-Resource Tasks** refer to problems (usually on the specific domains) where there is a limited amount of data available. **(B) Existing Data Augmentation** approaches expand the seed data with itself (policy for FMLA), which results in the limited diversity of the generated data samples (the same FMLA policy). **(C) Our Retrieval-Augmented Data Augmentation (RADA)** framework generates the new data with the external context (concurrent usage of FMLA and paternity leave), retrieved from the external datasets, along with the seed data, yielding more diverse and useful samples (paternity leave). **(Upper Right:)** Our RADA outperforms existing data augmentation methods, demonstrating the quality of generated samples. **(Lower Right:)** The generated data samples from RADA are more diverse than existing data augmentation, based on the t-SNE visualization.

sources accumulated in existing data pools, which can be utilized for data augmentation. Moreover, by leveraging the contextual understanding capabilities of LLMs, we can effectively utilize a mixture of samples drawn from the initial seed data, other datasets, or a combination of both. This can enable the synthesis of new samples, which mirror the characteristics of the seed data while being diverse.

However, not all samples from external datasets are useful for data augmentation, as most of them may not align with the characteristics of the seed data. Thus, inspired by the motivation to use external data instances while overcoming the problem of many of their irrelevancies, in this work, we propose a novel LLM-powered Retrieval-Augmented Data Augmentation (RADA) framework (See Figure 1, (C)). Specifically, the input of our data augmentation approach consists of in-context examples containing example instances, along with a target context that elicits a new sample generation. To be more specific, for open-domain question answering, which aims to answer a question based on information in a document, a sequence of multiple triplets of the document, question, and answer is used for in-context, while the target context is the document from which new question-answer pairs are generated. Then, our RADA flexibly employs multiple retrieval strategies to construct these in-context and target-context with samples from both original and external datasets, enabling diverse data augmentation, unlike the conventional approaches that rely solely on the initial seed data.

We validate the effectiveness of RADA in augmenting low-resource datasets on multiple domain-specific datasets, where we consider both the training and test-time data augmentation scenarios. The experimental results show that RADA consistently surpasses several LLM-powered data augmentation baselines on all datasets. In addition, a key finding from our analyses is the dual benefit offered by our RADA: the incorporation of external data sources enhances the diversity of the generated instances, while the retrieval mechanism ensures maintaining their semantic alignment with the initial seed data.

Our findings and contributions are threefolds:

- We point out the limitation of existing data augmentation approaches that rely on initial seed data alone, leading to a lack of diversity.
- We introduce a novel retrieval-augmented data augmentation framework, which performs retrieval over external data sources to generate diverse data based on information within and across the original and retrieved samples.
- We validate our RADA in augmenting data on low-resource settings with training and test-time scenarios, demonstrating its efficacy in generating the diverse and high-quality data.

2 Related Work

2.1 Large Language Models

Large Language Models (LLMs), trained on vast amounts of textual corpora with multiple training strategies along with a large number of parameters, have demonstrated remarkable capability of handling diverse tasks (Brown et al., 2020; Touvron et al., 2023; OpenAI, 2023; Anil et al., 2023). A notable feature of these models is their ability to perform in-context learning, which means they can understand and learn from examples or instructions provided in the input and then adapt their responses based on this information, without requiring retraining for each specific task (Brown et al., 2020; Wei et al., 2022; Min et al., 2022; Chen et al., 2022). Due to its simplicity yet effectiveness and versatileness, several approaches have been introduced to improve the quality of the LLM context. In particular, Lyu et al. (2023) constructs pseudo-demonstrations, for the case where examples in the context are unavailable, by retrieving relevant instances from the external corpus based on their similarities with the input query. Similarly, Ram et al. (2023) and Baek et al. (2023) augment LLMs by prepending relevant documents or facts retrieved from the external corpus in their input context, to improve the factuality of LLM responses. Lastly, Long et al. (2023) targets adapting LLMs with in-context examples (which are adaptively retrieved) for domain adaptation. However, existing works do not focus on augmenting the data based on the retrieval of its relevant samples from other datasets, through in-context learning of LLMs.

2.2 Data Augmentation

Despite the notable successes of LLMs, their performance significantly deteriorates in low-resource settings, particularly for domain-specific environments where the data available for training is very scarce (for instance, in the case of emerging events like novel viruses) or, in certain cases, completely unavailable (such as in privacy-sensitive enterprise contexts) (Ling et al., 2023; Chen et al., 2023b; Baldazzi et al., 2023). Further, they are less likely to be trained with ones similar to these specialized data, leading to constrained capability in handling them. To address this challenge, numerous studies have proposed to expand the original seed data with various data augmentation techniques (Feng et al., 2021; Li et al., 2022). Early works utilized token-level perturbation approaches, which either alter texts (Sahin and Steedman, 2018; Wei and Zou, 2019b) or interpolate them (Chen et al., 2020; Guo et al., 2020). Recent studies have shifted the focus towards utilizing the capability of generative language models, since they may internalize the useful knowledge to generate samples relevant to the seed data. Previous works on this line trained relatively smaller language models, based on the input-output pairs of the seed data to generate new outputs from the input variants (Yao et al., 2018; Anaby-Tavor et al., 2020; Lee et al., 2020). Also, more recent works have used LLMs, which have much greater capability in generating high-quality data (sometimes surpassing human-level performances) without requiring task-specific training (Honovich et al., 2023a; Whitehouse et al., 2023; Lee et al., 2023). Specifically, in information retrieval, some studies have generated synthetic queries with LLMs, to match the unlabeled documents with them (Bonifacio et al., 2022; Dai et al., 2023b; Saad-Falcon et al., 2023). Similarly, some other studies have proposed LLM-powered methods for specific downstream tasks, such as text classification (Dai et al., 2023a; Sahu et al., 2023), reading comprehension (Samuel et al., 2023), or multi-hop question answering (Chen et al., 2023c). This trend also goes to empowering the collection of instruction-tuning and alignment datasets for LLM training, which expands actual data samples with synthetic samples generated from LLMs themselves (Honovich et al., 2023b; Wang et al., 2023a,b; Li et al., 2023). However, in the low-resource setting, the seed data samples available to use for data augmentation are extremely scarce, which may result in suboptimal quality and limited diversity of the generated data. In this work, we propose to overcome this limitation by augmenting the data generation process with retrieval from larger external samples.

3 Methodology

In this section, we present a Retrieval-Augmented Data Augmentation (RADA) framework.

3.1 Problem Statement

We begin with introducing the problem of domain-specific tasks under low-resource settings, followed by describing LLMs for data augmentation.

Low-Resource Domain-Specific Tasks Before explaining the low-resource tasks that we focus on, we define conventional natural language tasks. Formally, their goal is to predict a label $\boldsymbol{y}$ given an input $\boldsymbol{x}$, where $\boldsymbol{x}$ and $\boldsymbol{y}$ are comprised of a sequence of tokens: $\boldsymbol{x} = [x_1, x_2, ..., x_{|\boldsymbol{x}|}]$ and $\boldsymbol{y} = [y_1, y_2, ..., y_{|\boldsymbol{y}|}]$. Then, the training data $\mathcal{D}$ can be represented as an aggregation of input-output pairs: $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{N}$ where its size N can vary widely from just a few dozens to several millions.

In this work, we target handling challenging scenarios where N is notably small, usually referred to as low-resource settings. These settings are particularly prevalent in domain-specific tasks (within legal, medical, or technical fields), where the availability of labeled data is inherently limited due to the specialized nature of the domain or the scarcity of domain experts for annotation; however, its quality and size are crucial to train performant models.

LLMs for Data Augmentation A typical way to handle the low-resource domain tasks is to expand the training data $\mathcal{D}$ with data augmentation techniques, which has been recently powered by LLMs due to their strong text-generation capabilities. Formally, let us first describe the LLM as a model parameterized by θ, which takes the input $\boldsymbol{x}$ and then generates the output $\boldsymbol{y}$, represented as follows: $\boldsymbol{y} = \texttt{LLM}_\theta(\boldsymbol{x})$. Here, θ is trained with massive text corpora with several training strategies and, after that, it usually remains fixed due to the costs of further training. Also, $\boldsymbol{x}$ can be any form of text, referred to as a prompt, which includes task-dependent instructions and contexts (such as demonstrations), to guide LLMs in generating outputs that align with the user's intent, which is data augmentation in our work, discussed below.

The primary goal of data augmentation is to expand the diversity and amount of data $\mathcal{D}$ available for model training (and for testing in certain use cases such as test-time adaption), without manually collecting the new data, for tackling specific tasks especially on low-resource domains. Formally, this data augmentation process can be represented as follows: $\mathcal{D}' = f(\mathcal{D})$, where f is the model (or technique) designed to generate new input-output pairs $(\boldsymbol{x}', \boldsymbol{y}')$ for the augmented dataset $\mathcal{D}'$, which is achieved by leveraging the underlying patterns, contexts, and knowledge existing in seed data $\mathcal{D}$. However, while there have been great successes in advancing the augmentation methods f in several different ways, for example, training the generative models or further prompting LLMs with the given original data, they mainly focus on expanding the original data $\mathcal{D}$ with itself. On the other hand, we can potentially incorporate any external sources of information easily available at hand, which could introduce greater diversity and quality in generating the samples for data augmentation. In addition, especially in low-resource settings, the available data to use as a source for expansion is largely scarce, which poses a significant challenge as the augmentation method f is operationalized with only limited samples, leading to the generation of samples that may lack the desired diversity and quality.

3.2 Retrieval-Augmented Data Augmentation

To tackle the aforementioned drawbacks of existing data augmentation approaches, we propose a novel data augmentation method (from a different angle), that leverages available external datasets.

Data Generation with External Resources We redefine the concept of previous data augmentation to incorporate samples from external resources, represented as follows: $\mathcal{D}' = f(\mathcal{D}, \mathcal{C})$ where $\mathcal{C}$ is an external data store that is composed of input-output pairs $(\boldsymbol{x}, \boldsymbol{y})$ aggregated from all available datasets. Notably, among the options to instantiate f, we follow a recent trend that uses LLMs with prompting, to harness their capabilities in understanding the longer and complex context (to jointly consider multiple samples from different datasets). This is not easily achievable by traditional smaller models without additional labeling for and excessive training on them. Yet, the different challenge lies not only in the limitation that not all the external data samples can be accommodated within the context length of LLMs, but also in the fact that many of these samples may not be pertinent for generating valuable augmentations for $\mathcal{D}$. Therefore, addressing these critical issues necessitates answering the question: How can we selectively integrate only the pertinent instances from the extensive data store $\mathcal{C}$?

3.2.1 Retrieving Relevant Instances

We now turn to answer the question of retrieving contextually relevant instances from the data store $\mathcal{C}$, which is critical as it ensures that the data produced by LLMs is not only diverse and high-quality but also contextually coherent and aligned with the nuances of the target dataset $\mathcal{D}$. In the following, we first provide the general formulation of the retrieval and then propose our two specific instantiations of the retrieval for data augmentation.

Formally, for a given input instance $\boldsymbol{q}$, the goal of a retriever is to identify and fetch a ranked list of k entries from a large corpus, which are deemed most relevant to the input, represented as follows: $\{\boldsymbol{c}_i\}_{i=1}^k = \texttt{Retriever}(\boldsymbol{q}, \mathcal{C})$ where $\boldsymbol{c}_i \in \mathcal{C}$. Here, $\boldsymbol{q}$ can be a textual query; $\mathcal{C}$ is the corpus (which is typically a large collection of documents) from which information is to be retrieved; `Retriever` is designed with keyword-based search algorithms or neural embedding-based models (Robertson et al., 1994; Karpukhin et al., 2020).

It is worth noting that, unlike typical retrieval approaches that primarily focus on sourcing relevant documents that are likely to contain the answers to the given query, in the context of our retrieval-augmented data augmentation scenario, we aim at fetching the relevant instances from other datasets, which are used as a source for generating the data along with the original samples. Therefore, these retrieved instances should ideally facilitate the generation of new and enriched samples. In addition, the instances to be retrieved can vary, which can

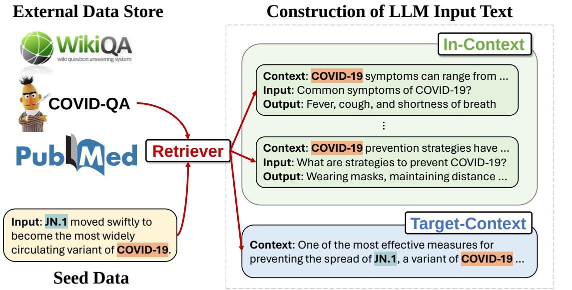

Figure 2: **RADA Framework Overview**. We first retrieve the external instances (relevant to the seed data) from the external data store, and construct in-context and target-context of LLM prompts with the retrieved samples along with the seed data.

be either complete input-output pairs or simply the inputs or outputs alone, depending on the specific requirements of data augmentation processes. We explain how we design retrieval in Section 3.2.2.

3.2.2 Retrieval for Data Augmentation

The input to LLMs can be viewed from two different perspectives: in-context learning which refers to their ability to learn from the input demonstrations; and task-solving where the model executes specific tasks requested by users (e.g., data augmentation). According to them, we propose two distinct instantiations of retrieval for LLM-powered data augmentation below (illustrated in Figure 2).

Retrieval for In-Context Learning In-context learning plays a crucial role in enabling LLMs to align their outputs with the contextual cues provided in the input examples. Similarly, in the context of data augmentation, it may enable LLMs to learn from examples (e.g., input-output pairs) in the seed data, to generate new input-output pairs. However, in low-resource settings that we consider, the combination of data samples to provide as the examples in the input prompt is largely limited. This limitation highlights the advantage of our retrieval-augmented data augmentation framework, which can fill the input demonstrations with samples from external datasets. Yet, as not all the samples are relevant, we retrieve only the relevant samples based on the similarity between the sample in seed data $\mathcal{D}$ and the external sample in data store $\mathcal{C}$, as follows: $\{\boldsymbol{c}_i\}_{i=1}^k = \texttt{Retriever}(\boldsymbol{q}, \mathcal{C})$ where $\boldsymbol{q} \in \mathcal{D}$[1]. Mathematically, the combination of demonstrations to use as the LLM input is expanded to $O((k \times |\mathcal{D}|)^3)$ from $O(|\mathcal{D}|^3)$, where $|\mathcal{D}|$ is typically small in the low-resource setting and we assume using 3 demonstrations with top-k sample retrievals.

[1]The similarity calculation mechanism can vary, and, in this work, we consider the similarity between input queries.

Retrieval for Target Sample Generation Unlike in-context examples providing background information for data augmentation, the context to be retrieved and used here has a different goal, which should serve as a source for generating a complete input-output pair or one among them when given the other, depending on specific use cases. Specifically, a certain document can be used as a context to derive a query-answer pair, along with their in-context examples. Another example is to provide a question as a context and then generate its answers, or vice versa to augment queries. It is worth noting that, while the usage of instances from the store $\mathcal{C}$ is different, their retrieval mechanism is the same as how we retrieve instances for in-context examples. Formally, $\{\boldsymbol{c}_i\}_{i=1}^k = \texttt{Retriever}(\boldsymbol{q}, \mathcal{C})$ where $\boldsymbol{q}$ can be either the document or the question from $\mathcal{D}$. Also, the augmented samples generated directly from the retrieved instances are similar in nature to the original samples, as we consider relevant top-k instances, ensuring a high degree of contextual coherence with seed samples while being more diverse against the generation with seed.

4 Experimental Setups

In this section, we outline the experimental setups. We provide additional details in Appendix A.

4.1 Tasks and Datasets

We validate our RADA on training data augmentation and test-time data augmentation scenarios.

Training Data Augmentation The goal of training data augmentation is to expand the given samples, which is useful when new events occur that the model needs to adapt to, while having only limited data available for training. To test RADA with this scenario, we use three low-resource domain-specific datasets: Covid QA (Möller et al., 2020) that is annotated by medical doctors for tackling the COVID-19 pandemic; Policy QA (Ahmad et al., 2020) that is designed with specialized policies about website privacy; and Tech QA (Castelli et al., 2020) that is constructed with questions on technical public forums for the IT domain. In addition, to simulate the low-resource settings, we sample 10, 30, and 100 instances from the training dataset.

Test-Time Data Augmentation The assumption of test-time data augmentation is more challenging, considering the case where there is no data available for training due to strict privacy concerns (e.g., users or institutions may not want to share their

Table 1: **Training data augmentation results** with T5-base as the base model for training. In the second row, 10, 30, and 100 denote the number of initial seed data. We emphasize the statistically significant results under the t-test of p < 0.05 in bold.

	Covid QA			Policy QA			Tech QA			Average		
Methods	10	30	100	10	30	100	10	30	100	10	30	100
Seed Data	57.07	66.93	68.97	6.25	16.26	28.09	12.28	17.59	33.90	25.2	33.59	43.65
Augment w/ Seed Data	62.74	64.69	65.01	28.08	27.49	25.89	40.20	42.07	42.42	43.67	44.75	44.44
Self-Instruct	63.34	61.90	64.20	27.48	27.50	27.53	33.20	39.13	37.55	41.34	42.84	43.09
QA Generation	51.72	48.98	39.05	20.04	20.46	20.95	30.01	30.99	32.80	33.92	33.48	30.93
CQA Generation	67.00	67.01	67.80	27.30	24.96	25.94	28.08	30.94	31.88	40.79	40.97	41.87
Seed + External Data	62.30	62.81	63.50	25.72	25.60	29.34	34.82	35.46	37.06	40.95	41.29	43.30
PAQ (non-LLM)	65.23	66.55	66.72	24.37	25.87	27.48	24.03	25.65	29.89	37.88	39.36	41.36
RADA (Ours)	**67.55**	**67.95**	68.36	**28.83**	**28.25**	28.88	40.44	**44.41**	**45.81**	**45.61**	**46.87**	**47.68**

private data to train models) (Jeong et al., 2023). For this scenario, we select and use three specific domains from the MMLU dataset (Hendrycks et al., 2021) as it does not have direct training instances (aligned with our validation purpose), as well as using previous Covid QA, Policy QA, and Tech QA with no training samples available for this setup.

External Resources for Retrieval We construct the external data store serving as a retrieval source by aggregating samples from other datasets. Specifically, for Covid QA, Policy QA, and Tech QA designed for open-domain Question Answering (QA), we use Natural Questions (NQ) (Kwiatkowski et al., 2019) and labeled subset (Xu et al., 2020) of MS MARCO (Nguyen et al., 2016), covering broad domains with questions asked on web search. For MMLU that targets multi-choice QA, we use its official auxiliary data collected from similar datasets.

4.2 Baselines and Our Model

We compare our approach to several LLM-powered data augmentation baselines to ensure a fair evaluation. Also, we include non-LLM-based approaches for reference purposes, contrasting them with LLM-based methods (See Appendix B for further discussion and results on them). 1. **Seed Data** – It uses only the seed data for training models without extra data augmentation steps. 2. **Augment w/ Seed Data** – It expands the seed data by generating new data instances from the seed data samples, where samples for in-context learning and target–context selection are randomly picked. 3. **Self--Instruct** – It (Wang et al., 2023a) aims to bootstrap new tasks only with limited seed examples, by incorporating the generated data instances in the data pool and leveraging them along with the seed data iteratively, where the samples in the pool are used to construct the in-context and target samples. 4. **CQA Generation** – It (Samuel et al., 2023) generates a context and then, based on it, subsequently generates a question-answer pair, where existing seed data samples are used for in-context learning. Its variant (**QA Generation**) generates a question-answer pair with in-context learning (Ye et al., 2022). 5. **Seed + External Data** – It trains the models with the seed data instances as well as all the instances available in the external data pool. 6. **PAQ (non-LLM)** It (Lewis et al., 2021) is a state-of-the-art non-LLM-based method, which selects passages, extracts answers, generates questions, and filters some of them, with conventional NER tools and smaller LM. 7. **RADA** – This is our model that generates samples by retrieving samples (relevant to the seed data) from the external corpus and using them for in-context and target context.

Table 2: **Test-time data augmentation results** on subdomains of MMLU and domain-specific QA datasets. We use Llama2-7B as the base model for MMLU and T5-base for others.

MMLU	**CS**	**Biology**	**Law**	**Average**
5-Shots w/ Training	32.00	47.74	64.46	48.07
External Data	48.00	54.52	66.12	56.21
RADA (Ours)	**49.00**	**55.48**	**70.25**	**58.24**

Domain-Specific QA	**Covid**	**Policy**	**Tech**	**Average**
External Data	54.02	19.32	12.97	28.77
PAQ (non-LLM)	61.22	25.03	19.83	35.36
RADA (Ours)	**66.03**	**29.14**	**29.17**	**41.45**

We note that, for the test-time data augmentation scenario, since the samples having complete input-output pairs are unavailable, we cannot compare against the baselines requiring in-context examples; yet, RADA can run with only the target context.

4.3 Implementation Details

We use Llama2-7B-Chat (Touvron et al., 2023) as the basis for data augmentation across all methods. For fine-tuning we use either T5-base (Raffel et al., 2020) or Llama2-7B, to measure the effectiveness of different approaches directly without worrying about data contamination as they are not trained on any downstream tasks/datasets. For the number of data augmented, unless otherwise stated, we produce samples amounting to 30 times that of the seed data and train models with the seed and generated data. A retriever used to retrieve instances is

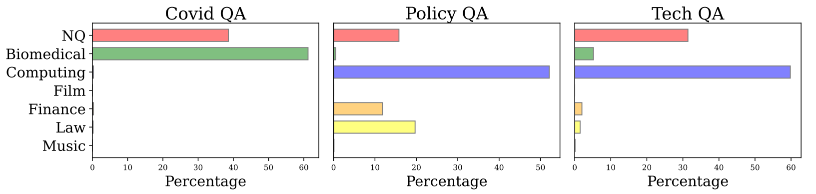

Figure 3: **Breakdown results of retrieved instances** on three domain-specific QA datasets, where samples in the retrieval pool are one of Biomedical, Computing, Film, Finance, Law, and Music domains, as well as NQ (which covers general domains).

Domains	Covid QA	Tech QA
All	67.55	40.44
Biomedical	**67.75**	40.09
Computing	66.70	**42.67**

Table 3: **Results of the hand-crafted data store**, selectively using only the most suitable external domain as the retrieval pool for domain-specific QA.

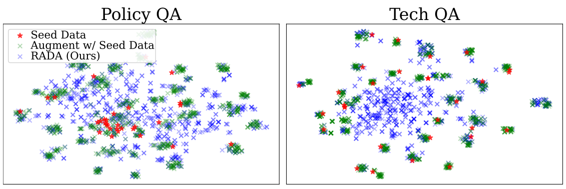

Figure 4: **Embedding-space visualization results of samples** including the seed data and augmented data, with t-SNE.

Figure 5: **Results of ROUGE-L score distributions** measured between the seed data and generated data on Tech QA.

DistilBert TAS-B (Hofstätter et al., 2021). We report results with the F1 score for Covid QA, Policy QA, and Tech QA datasets, and the accuracy for MMLU, following standard evaluation protocols. We provide prompts used to elicit data augmentation and answer generation in Appendix A.

5 Experimental Results

Main Results We conduct experiments on two different data augmentation scenarios and report the results of training data augmentation in Table 1[2] and the test-time augmentation results in Table 2 (See Table 9 and Table 10 for standard deviations). As shown in them, RADA substantially outperforms all baselines except for a few settings, while none of the baselines achieve statistically significant results, demonstrating the effectiveness of RADA. In addition, two particular superior points of baselines are not an unexpected result, since the number of initial seed data (100) is already large. Also, the baseline of Augment w/ Seed Data is further coupled with a large number of external data samples (117,580), which may provide sufficient information to handle the task, which is much larger than the data used for RADA (30,100). We note that the average score of the non-LLM-based PAQ approach is low, compared to LLM-based methods, which confirms the effectiveness of using LLMs for data augmentation perhaps thanks to their prior knowledge (See Appendix B for more results and discussion). Moreover, as shown in Table 2, RADA is highly effective in the challenging test-time data augmentation scenario (where no data is available for training), outperforming the model trained with all the external data instances. This may be due to our retrieval strategy, which results in generating samples that are relevant to the test data.

Analysis of Retrieval To understand which data instances are retrieved for data augmentation and what are their effectiveness, we conduct a comprehensive analysis. Firstly, we visualize the categories of retrieved instances for domain-specific QA in Figure 3, which shows that (mostly) only the relevant instances are retrieved and used for data augmentation for each specific task. For example, the Biomedical domain is the dominant field of retrieval source for Covid QA; meanwhile, the Computing domain is for Tech QA. In addition, to see the contribution of relevant retrieval, we restrict the retrieval domain to the one that is the most relevant to the given specific dataset. For example, we use only the Biomedical domain for Covid QA and the Computing domain for Tech QA. As shown in Table 3, we observe that when manipulating the retrieval pool, the performance further increases (as instances from irrelevant domains are not retrieved), which reaffirms the effectiveness of retrieval and its room for improvement for data augmentation.

Analysis of Augmented Data Diversity A notable advantage of RADA is that it intuitively can generate more diverse samples than what could be achieved by existing data augmentation approaches that use the seed data alone, by augmenting this process with the retrieval from external data samples. To measure this ability, we visualize the embedding space of the augmented samples across different models in Figure 4 and report their lexical overlaps in Figure 5. Specifically, for the visualization, we first embed the generated instances with Sentence-BERT (Reimers and Gurevych, 2019a) into the

[2] We observe that the performance of Llama2 even after fine-tuning on the seed data and the augmented data is much inferior to T5-base on domain-specific QA; thus, we report results for them with T5 and further discuss it in Appendix B.

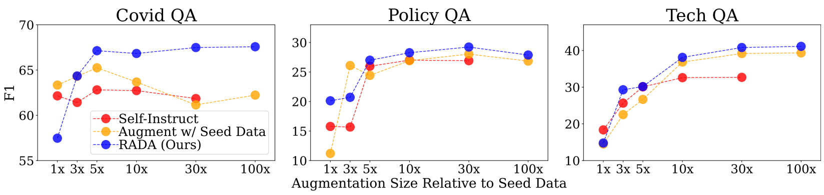

Figure 6: **Results of varying the augmentation size** on domain-specific QA, where we increase the size by factors of 1, 3, 5, 10, 30, and 100 relative to the seed data size.

Methods	Tech QA
RADA (Ours)	**44.41**
w/o In-context Retriever	41.24
w/o Target-context Retriever	34.42
w/o All Retrievers	30.38

Table 4: **Ablation study** of the proposed RADA on the Tech QA dataset.

latent space and project them with t-SNE (van der Maaten and Hinton, 2008). From this, we observe that, unlike Augment w/ Seed Data whose generated samples are close to the seed data, the samples generated from RADA are broadly dispersed across the space. Further, we measure the max ROUGE-L scores between the seed instances and the generated instances where lower scores indicate higher diversity. As shown in Figure 5, RADA generates distinct samples to the seed data thanks to retrieving and utilizing the external contexts beyond the seed data, unlike baselines that rely solely on it.

Analysis of Augmented Data Size To see how the performance changes as a function of the size of augmented data samples, we vary the augmentation size relative to the seed data size by a factor of 1, 3, 5, and up to 100 times and report the results in Figure 6[3]. Firstly, when the amount of augmented data is very small, baseline performances are comparable with RADA since the data samples that can be generated from the seed data alone can have a certain diversity level as we augment only a small amount. Yet, as the size of augmentation expands, RADA consistently outperforms baselines, showcasing its ability to generate broader and richer samples through retrieval augmentation, while the performance saturates after a 30-time increase.

Ablation Study To see how each component of RADA affects the overall performance, we conduct an ablation study where we replace our in-context and target-context retrieval modules with random retrievals. As shown in Table 4, we observe that, without retrieving relevant instances, the performances drop substantially since irrelevant samples (to the target tasks/datasets) are used to construct the in-context examples and target context, leading to generating the samples not useful for them. Furthermore, the target-context retriever is particularly important for data augmentation, since this context is used to directly derive the instances for training.

Analysis of Using Different LLMs Finally, we conduct an auxiliary analysis to see whether the

[3]Due to the cost of running Self-Instruct, we are not able to generate its samples for the 100 times augmentation-level.

Table 5: **Results of another LLM (ChatGPT)** for data augmentation on domain-specific QA with seed examples of 10.

	Covid	Policy	Tech	Average
Self-Instruct	57.86	26.20	33.42	39.16
CQA Generation	65.64	27.20	34.16	42.33
RADA (Ours)	**67.19**	**28.59**	**36.17**	**43.98**

superiority of RADA is consistent across different LLMs, compared to existing baselines. In particular, we use ChatGPT 3.5 (released on June 13, 2023) as the basis model for data augmentation, and report the results in Table 5. From this, we observe that RADA significantly outperforms baselines with another LLM, demonstrating its robustness across different LLMs for data augmentation.

6 Conclusion

In this work, we pointed out the limitation of existing data augmentation approaches that use the seed data alone for low-resource domain tasks, leading to generating suboptimal and less diverse instances, despite the existence of plenty of external samples available. Inspired by this, we proposed the LLM-powered Retrieval-Augmented Data Augmentation (RADA) framework, which augments the seed data by leveraging the samples retrieved from the external data store based on their relevance with the seed data, during data augmentation. Specifically, the input to LLMs for data augmentation can be viewed from two different angles of in-context examples and task-solving context, and we constructed them through samples from within and across the seed data and the retrieved data. Through extensive evaluation results on multiple datasets with training and test-time data augmentation scenarios, we showed that RADA outperforms strong LLM-powered data augmentation baselines substantially. In addition, our findings reveal that the data samples generated from our approach are much more diverse against baselines while being relevant to the seed data, due to leveraging retrieval for data augmentation. We believe that RADA will pave the way for enhancing the model performances on realistic low-resource domain-specific tasks, which have arisen as very important problems recently due to the limited availability and privacy concerns of data.

Limitations

In this section, we faithfully discuss some remaining room for improvements to our RADA framework. First of all, the effectiveness of our retrieval-augmentation approach (by its nature) depends on the quality and relevance of the external data store. Thus, the performance of RADA may degenerate if the retrieval source is not truly aligned with our seed data, and we leave exploring this new setting as future work. Also, investigating the scenario of continuously updating the retrieval pool over time would be interesting for future work as well. On the other hand, due to the heavy cost of fine-tuning LLMs, data sample efficiency (i.e., reducing the amount of samples to train while maintaining the model performance) becomes an important agenda. While we do have some preliminary results on filtering augmented samples in Appendix B, it would be interesting to developing more on this direction.

Ethics Statement

While our RADA is superior in generating more diverse and high-quality samples (compared to existing data augmentation approaches), its performance is not flawless: the retriever might retrieve offensive or harmful instances for data augmentation, and the generator might produce plausible yet factually incorrect instances. Therefore, it may be carefully used for mission-critical domains, such as biomedical or legal fields, (perhaps with the help of domain-experts during the augmentation process).

References

Wasi Uddin Ahmad, Jianfeng Chi, Yuan Tian, and Kai-Wei Chang. 2020. Policyqa: A reading comprehension dataset for privacy policies. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 743–749. Association for Computational Linguistics.

Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper, and Naama Zwerdling. 2020. Do not have enough data? deep learning to the rescue! In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7383–7390. AAAI Press.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul Ronald Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, and et al. 2023. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Jinheon Baek, Alham Fikri Aji, and Amir Saffari. 2023. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. *arXiv preprint arXiv:2306.04136*.

Teodoro Baldazzi, Luigi Bellomarini, Stefano Ceri, Andrea Colombo, Andrea Gentili, and Emanuel Sallinger. 2023. Fine-tuning large enterprise language models via ontological reasoning. In *Rules and Reasoning - 7th International Joint Conference, RuleML+RR 2023, Oslo, Norway, September 18-20, 2023, Proceedings*, volume 14244 of *Lecture Notes in Computer Science*, pages 86–94. Springer.

Luiz Henrique Bonifacio, Hugo Queiroz Abonizio, Marzieh Fadaee, and Rodrigo Frassetto Nogueira. 2022. Inpars: Unsupervised dataset generation for information retrieval. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 2387–2392. ACM.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Vittorio Castelli, Rishav Chakravarti, Saswati Dana, Anthony Ferritto, Radu Florian, Martin Franz, Dinesh Garg, Dinesh Khandelwal, J. Scott McCarley, Mike McCawley, Mohamed Nasr, Lin Pan, Cezar Pendus, John F. Pitrelli, Saurabh Pujar, Salim Roukos, Andrzej Sakrajda, Avirup Sil, Rosario Uceda-Sosa, Todd Ward, and Rong Zhang. 2020. The techqa dataset. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020,* *Online, July 5-10, 2020*, pages 1269–1278. Association for Computational Linguistics.

Angelica Chen, David M. Dohan, and David R. So. 2023a. Evoprompting: Language models for code-level neural architecture search. *arXiv preprint arXiv:2302.14838*.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. Mixtext: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2147–2157. Association for Computational Linguistics.

Meng Chen, Hongyu Zhang, Chengcheng Wan, Zhao Wei, Yong Xu, Juhong Wang, and Xiaodong Gu. 2023b. On the effectiveness of large language models in domain-specific code generation. *arXiv preprint arXiv:2312.01639*.

Mingda Chen, Xilun Chen, and Wen-tau Yih. 2023c. Efficient open domain multi-hop question answering with few-shot data synthesis. *arXiv preprint arXiv:2305.13691*, abs/2305.13691.

Mingda Chen, Jingfei Du, Ramakanth Pasunuru, Todor Mihaylov, Srini Iyer, Veselin Stoyanov, and Zornitsa Kozareva. 2022. Improving in-context few-shot learning via self-supervised training. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 3558–3573. Association for Computational Linguistics.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Wei Liu, Ninghao Liu, et al. 2023a. Auggpt: Leveraging chatgpt for text data augmentation. *arXiv preprint arXiv:2302.13007*.

Zhuyun Dai, Vincent Y. Zhao, Ji Ma, Yi Luan, Jianmo Ni, Jing Lu, Anton Bakalov, Kelvin Guu, Keith B. Hall, and Ming-Wei Chang. 2023b. Promptagator: Few-shot dense retrieval from 8 examples. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*, abs/2305.14314.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard H. Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 968–988. Association for Computational Linguistics.

Arnav Gudibande, Eric Wallace, Charlie Snell, Xinyang Geng, Hao Liu, Pieter Abbeel, Sergey Levine, and Dawn Song. 2023. The false promise of imitating proprietary llms. *arXiv perprint arXiv:2305.15717*.

Biyang Guo, Yeyun Gong, Yelong Shen, Songqiao Han, Hailiang Huang, Nan Duan, and Weizhu Chen. 2022. Genius: Sketch-based language model pre-training via extreme and selective masking for text generation and augmentation. *ArXiv*, abs/2211.10330.

Demi Guo, Yoon Kim, and Alexander M. Rush. 2020. Sequence-level mixed sample data augmentation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 5547–5552. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 113–122. ACM.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2023a. Unnatural instructions: Tuning language models with (almost) no human labor. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 14409–14428. Association for Computational Linguistics.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2023b. Unnatural instructions: Tuning language models with (almost) no human labor. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 14409–14428. Association for Computational Linguistics.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C. Park. 2023. Test-time self-adaptive small language models for question answering. *ArXiv*, abs/2310.13307.

Akbar Karimi, L. Rossi, and Andrea Prati. 2021. Aeda: An easier data augmentation technique for text classification. In *Conference on Empirical Methods in Natural Language Processing*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the*

2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7:452–466.

Dong Bok Lee, Seanie Lee, Woo Tae Jeong, Donghwan Kim, and Sung Ju Hwang. 2020. Generating diverse and consistent QA pairs from contexts with information-maximizing hierarchical conditional vaes. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 208–224. Association for Computational Linguistics.

Dong-Ho Lee, Jay Pujara, Mohit Sewak, Ryen White, and Sujay Kumar Jauhar. 2023. Making large language models better data creators. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 15349–15360. Association for Computational Linguistics.

Patrick Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Kuttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. 2021. Paq: 65 million probably-asked questions and what you can do with them. *Transactions of the Association for Computational Linguistics*, 9:1098–1115.

Bohan Li, Yutai Hou, and Wanxiang Che. 2022. Data augmentation approaches in natural language processing: A survey. *AI Open*, 3:71–90.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Luke Zettlemoyer, Omer Levy, Jason Weston, and Mike Lewis. 2023. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*, abs/2308.06259.

Chen Ling, Xujiang Zhao, Jiaying Lu, Chengyuan Deng, Can Zheng, Junxiang Wang, Tanmoy Chowdhury, Yun-Qing Li, Hejie Cui, Xuchao Zhang, Tian yu Zhao, Amit Panalkar, Wei Cheng, Haoyu Wang, Yanchi Liu, Zhengzhang Chen, Haifeng Chen, Chris White, Quanquan Gu, Jian Pei, Carl Yang, and Liang Zhao. 2023. Domain specialization as the key to make large language models disruptive: A comprehensive survey.

Quanyu Long, Wenya Wang, and Sinno Jialin Pan. 2023. Adapt in contexts: Retrieval-augmented domain adaptation via in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 6525–6542. Association for Computational Linguistics.

Kai Lv, Yuqing Yang, Tengxiao Liu, Qinghui Gao, Qipeng Guo, and Xipeng Qiu. 2023. Full parameter fine-tuning for large language models with limited resources. *arXiv preprint arXiv:2306.09782*.

Xinxi Lyu, Sewon Min, Iz Beltagy, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Z-ICL: zero-shot in-context learning with pseudo-demonstrations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 2304–2317. Association for Computational Linguistics.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2022. Metaicl: Learning to learn in context. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 2791–2809. Association for Computational Linguistics.

Timo Möller, Anthony Reina, Raghavan Jayakumar, and Malte Pietsch. 2020. Covid-qa: A question answering dataset for covid-19.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

OpenAI. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *arXiv preprint arXiv:2302.00083*, abs/2302.00083.

Nils Reimers and Iryna Gurevych. 2019a. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019b. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. In *Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, November 2-4, 1994*, volume 500-225 of *NIST Special Publication*, pages 109–126. National Institute of Standards and Technology (NIST).

Jon Saad-Falcon, Omar Khattab, Keshav Santhanam, Radu Florian, Martin Franz, Salim Roukos, Avirup Sil, Md. Arafat Sultan, and Christopher Potts. 2023. UDAPDR: unsupervised domain adaptation via LLM prompting and distillation of rerankers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 11265–11279. Association for Computational Linguistics.

Gözde Gül Sahin and Mark Steedman. 2018. Data augmentation via dependency tree morphing for low-resource languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 5004–5009. Association for Computational Linguistics.

Gaurav Sahu, Olga Vechtomova, Dzmitry Bahdanau, and Issam Hadj Laradji. 2023. Promptmix: A class boundary augmentation method for large language model distillation. *arXiv preprint arXiv:2310.14192*.

Vinay Samuel, Houda Aynaou, Arijit Ghosh Chowdhury, Karthik Venkat Ramanan, and Aman Chadha. 2023. Can llms augment low-resource reading comprehension datasets? opportunities and challenges. *arXiv preprint arXiv:2309.12426*, abs/2309.12426.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663*, abs/2104.08663.

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, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023a. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 13484–13508. Association for Computational Linguistics.

Yue Wang, Xinrui Wang, Juntao Li, Jinxiong Chang, Qishen Zhang, Zhongyi Liu, Guannan Zhang, and Min Zhang. 2023b. Harnessing the power of david against goliath: Exploring instruction data generation without using closed-source models. *arXiv preprint arXiv:2308.12711*, abs/2308.12711.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Jason Wei and Kai Zou. 2019a. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. In *Conference on Empirical Methods in Natural Language Processing*.

Jason W. Wei and Kai Zou. 2019b. EDA: easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 6381–6387. Association for Computational Linguistics.

Chenxi Whitehouse, Monojit Choudhury, and Alham Fikri Aji. 2023. Llm-powered data augmentation for enhanced crosslingual performance. *arXiv preprint arXiv:2305.14288*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 38–45. Association for Computational Linguistics.

Ying Xu, Xu Zhong, Antonio José Jimeno-Yepes, and Jey Han Lau. 2020. Forget me not: Reducing catastrophic forgetting for domain adaptation in reading comprehension. In *2020 International Joint Conference on Neural Networks, IJCNN 2020, Glasgow, United Kingdom, July 19-24, 2020*, pages 1–8. IEEE.

Kaichun Yao, Libo Zhang, Tiejian Luo, Lili Tao, and Yanjun Wu. 2018. Teaching machines to ask questions. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4546–4552. ijcai.org.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. Zerogen: Efficient zero-shot learning via dataset generation. *arXiv preprint arXiv:2202.07922*.

Figure 7: **Results of ROUGE-L score distributions** measured between the seed data and generated data on Covid QA.

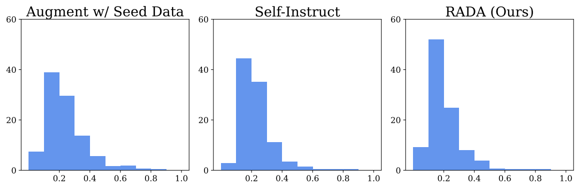

Figure 8: **Results of ROUGE-L score distributions** measured between the seed data and generated data on Policy QA.

A Additional Experimental Setups

Fine-tuning Details We provide more details on how to fine-tune models on the seed and augmented data samples. Firstly, for T5-base, we train it over 5 epochs with a batch size of 8 and a learning rate of 3×10^{-5}, selecting the best epoch to report the performance with inference. For Llama-7B, to train it with our computational resources available, we use the QLORA (Dettmers et al., 2023) technique, on which we use the epoch size of 30, the batch size of 1, and the learning rate of 2×10^{-4}. Lastly, we report the fine-tuning results with three runs.

Prompts The prompt used to elicit the data augmentation is provided in Table 12. For the domain-specific datasets including Covid QA, Policy QA, and Tech QA, we use the following prompt to generate the answer: "Context: { } Question: { } Answer: ". For the MMLU dataset, we use the following prompt: "Question: { } Answer Options: { } Answer:" where 5-shot examples prepended are the same as the one in the official code repository[4].

Computational Resources and Time We train and inference all baselines and our model by using one of the TITAN RTX, NVIDIA GeForce RTX 3080, NVIDIA GeForce RTX 3090, NVIDIA RTX A4000, NVIDIA RTX A5000, and Quadro RTX 8000 GPUs, depending on their availability at the time of run. The time required for training RADA ranges from a few minutes to about one and half day, which also depends on the number of the augmented data used for model fine-tuning.

Deep Learning Libraries In our experiments, we utilize the deep learning libraries as follows: PyTorch (Paszke et al., 2019), Transformers (Wolf et al., 2020), SentenceTransformers (Reimers and Gurevych, 2019b), and BEIR (Thakur et al., 2021). We will release the specific requirements for reproducing our results, upon releasing the code.

[4] https://github.com/hendrycks/test

Table 6: The average ROUGE-scores between the original data samples and the augmented data samples.

	Covid	Policy	Tech
Augment w/ Seed Data	0.34	0.29	0.39
Self-Instruct	0.33	0.28	0.32
RADA (Ours)	**0.30**	**0.25**	**0.24**

Table 7: Training time results on Covid QA, where we use T5 and Llama as the base for fine-tuning on augmented data.

# of seed	Bases	0-shot	5-shot	Seed	RADA (Ours)
10	T5	N/A	N/A	**53.94**	**67.49**
	Llama2	12.79	16.43	50.62	56.50
30	T5	N/A	N/A	**66.50**	**68.15**
	Llama2	12.79	16.43	55.48	53.62

B Additional Experimental Results

More Analysis of Data Diversity In addition to the result of ROUGE-L score distributions on Tech QA in Figure 5, we provide results on Covid QA and Policy QA in Figure 7 and Figure 8, respectively. From this, we consistently observe that the proposed RADA generates diverse instances during data augmentation, compared to other baselines. In addition, we provide more quantitative results reporting the average of ROUGE-scores between the original data samples and the augmented data samples in Table 6, reaffirming the advantage of our RADA in generating more diverse samples.

Results of Llama on Domain-Specific QA Here we discuss the training data augmentation results of Llama on domain-specific QA data (such as Covid QA). Specifically, in Table 7, we report its 0-shot and 5-shot performances, as well as its fine-tuning performances on seed data and augmented data. As shown in Table 7, despite the large number of parameters that Llama2-7B has (which is ten times larger than T5), we observe that Llama2 is inferior to T5. We conjecture that this may be because the general massive corpus used to pre-train Llama2 has little (to no) overlap or relevance with instances in domain-specific tasks. In other words, eliciting the domain-specific ability of Llama2 with fine-tuning may be largely suboptimal, when it does not have internalized knowledge about its corre-

Table 8: **Results of various filtering mechanisms** on domain-specific QA datasets with training data augmentation settings.

	Covid QA			Policy QA			Tech QA			Average		
Methods	10	30	100	10	30	100	10	30	100	10	30	100
RADA (Ours)	**67.49**	**68.15**	**68.57**	**29.23**	**28.49**	**29.18**	**40.81**	44.37	**46.93**	**45.84**	**47.00**	**48.23**
w/ ROUGE-based Filtering	66.21	67.25	66.84	28.35	28.09	28.31	37.75	**44.64**	46.74	44.10	46.66	47.30
w/ Embedding-based Filtering	67.19	67.67	67.27	28.62	28.13	28.65	40.02	**44.64**	46.74	45.27	46.82	47.55
w/o Answer Filtering	66.78	66.65	67.09	28.78	28.44	29.12	40.55	42.43	42.56	45.37	45.84	46.26

Table 9: Training data augmentation results where we report the standard deviations in parentheses and the statistically significant results (under the t-test of p-value < 0.05) in bold.

	Covid QA			Policy QA			Tech QA		
Methods	10	30	100	10	30	100	10	30	100
Seed Data	57.07 (2.76)	66.93 (0.38)	68.97 (0.46)	6.25 (1.21)	16.26 (3.46)	28.09 (0.49)	12.28 (2.37)	17.59 (0.48)	33.90 (2.34)
Augment w/ Seed Data	62.74 (1.41)	64.69 (0.01)	65.01 (0.51)	28.08 (0.41)	27.49 (0.47)	25.89 (0.16)	40.20 (0.92)	42.07 (1.52)	42.42 (1.01)
Self-Instruct	63.34 (1.58)	61.90 (0.18)	64.20 (0.24)	27.48 (0.53)	27.50 (0.13)	27.53 (0.27)	33.20 (0.75)	39.13 (0.76)	37.55 (0.53)
QA Generation	51.72 (1.15)	48.98 (1.82)	39.05 (1.91)	20.04 (0.77)	20.46 (0.55)	20.95 (0.22)	30.01 (0.13)	30.99 (0.23)	32.80 (0.78)
CQA Generation	67.00 (0.32)	67.01 (0.18)	67.80 (0.17)	27.30 (0.26)	24.96 (0.17)	25.94 (0.70)	28.08 (0.92)	30.94 (0.68)	31.88 (0.95)
Seed + External Data	62.30 (0.44)	62.81 (0.28)	63.50 (0.55)	25.72 (0.41)	25.60 (1.07)	29.34 (0.12)	34.82 (0.21)	35.46 (0.94)	37.06 (0.02)
PAQ (non-LLM)	65.23 (0.66)	66.55 (0.24)	66.72 (0.47)	24.37 (0.18)	25.87 (0.60)	27.48 (0.46)	24.03 (0.48)	25.65 (1.39)	29.89 (0.35)
RADA (Ours)	**67.55 (0.15)**	**67.95 (0.20)**	68.36 (0.25)	**28.83 (0.37)**	**28.25 (0.21)**	28.88 (0.50)	40.44 (0.53)	**44.41 (0.45)**	**45.81 (0.97)**

Table 10: Test-time data augmentation results where we report the standard deviations in parentheses and the statistically significant results (under the t-test of p-value < 0.05) in bold.

Domain-Specific QA	**Covid**	**Policy**	**Tech**
External Data	54.02 (0.42)	19.32 (0.11)	12.97 (0.52)
PAQ (non-LLM)	61.22 (0.22)	25.03 (0.34)	19.83 (0.83)
RADA (Ours)	**66.03 (0.15)**	**29.14 (0.18)**	**29.17 (0.98)**

sponding domain-specific tasks. In addition, this result may further highlight the fact that not all the larger models perform always better than the smaller models in low-resource settings, which gives us a promise to take advantage of computational efficiency, especially when dealing with extreme domain-specific tasks, or that specific LLMs may be required to handle each specific domain.

Results with Filtering We try various filtering approaches on the augmented data to fine-tune models with only the samples of high quality. Specifically, to further promote diversity in the generated samples from our RADA, we filter samples if they are similar to the already generated samples, based on their ROUGE scores or their embedding-level distances. Then, as shown in Table 8, these filtering techniques do not improve the model performance. This may further strengthen our claim that the augmented instances from RADA are already very diverse but also relevant to the seed data, which does not necessitate additional filtering mechanisms. On the other hand, if we relax the assumption that the passage should include the answer to the question for domain-specific QA, and subsequently do not apply the filtering strategy (checking the inclusiveness), the performance drops slightly in Table 8.

Table 11: Comparison results of our LLM-powered RADA approach against non-LLM-based methods on the challenging TechQA dataset, with the training time augmentation scenario. We report the standard deviations in parentheses and the statistically significant results (under the t-test) in bold.

	10	**30**	**100**
PAQ	24.03 (0.48)	25.65 (1.39)	29.89 (0.35)
GENIUS	12.28 (2.37)	26.90 (0.50)	43.55 (0.45)
EDA	38.27 (0.53)	41.93 (0.26)	45.21 (0.64)
AEDA	38.86 (0.30)	41.98 (0.30)	45.24 (0.16)
RADA (Ours)	**40.44 (0.53)**	**44.41 (0.45)**	**45.81 (0.97)**

More Results of Non-LLM-based Baselines It is worth noting that making a comparison of LLM-based approaches (including our RADA) over non-LLM-based methods is unfair since different LMs have different capabilities in generating outputs, which leads to far different quality of augmented samples. Therefore, to ensure a fair comparison across all data augmentation approaches, we set Llama2 as the basis for data augmentation. Nevertheless, to see the efficacy of non-LLM-based approaches, we compare our RADA against several recent and popular (non-LLM-based) methods, namely PAQ (Lewis et al., 2021), GENIUS (Guo et al., 2022), EDA (Wei and Zou, 2019a), and AEDA (Karimi et al., 2021), on the most challenging dataset (TechQA) that we observe in Table 1. Then, we report the results in Table 11. From this, we observe that RADA significantly outperforms previous non-LLM-based methods, demonstrating the effectiveness of using the LLM-based approach for data augmentation under low-resource settings, which may be due to LLM's prior knowledge.

Quantitative Analysis In Table 13, 14, 15, we provide examples of the augmented instances across different methods on Covid QA, Policy QA, and Tech QA. A key finding from these results is that the existing approach that uses only the seed data results in a limited diversity of generated samples, unlike our RADA which generates distinct yet contextually coherent samples with the seed data, thanks to the retrieval of relevant external samples.

Table 12: A list of prompts that we use for data augmentation with the proposed RADA framework. It is worth noting that the variable inside the parentheses { } is replaced with its actual string (e.g., context, question, answer options, and answer). Also, the last sentence of the prompt represents the target context, which is used as the main source of information to generate the augmented instance. For MMLU, we use the combinations of Version 1 and Version 2 for data augmentation.

Types	**Prompts**
Domain-specific QA	I want you to act as a question and answer generator. Your goal is to create an extractive question-answer pair based on a given context. The answer to the question must be a specific span from the given context. Context: {context 1} Question: {question 1} Answer: {answer 1} Context: {context 2} Question: {question 2} Answer: {answer 2} Context: {context 3} Question: {question 3} Answer: {answer 3} Context: {context}
MMLU (Version 1)	I want you to act as an answer options and answer generator. Your goal is to create four answer options and the answer pair based on a given question. The answer must be one of the generated answer options. Question: {question 1} Answer Options: {answer options 1} Answer: {answer 1} Question: {question 2} Answer Options: {answer options 2} Answer: {answer 2} Question: {question 3} Answer Options: {answer options 3} Answer: {answer 3} Question: {question}
MMLU (Version 2)	I want you to act as a question and answer generator. Your goal is to create an extractive question-answer pair based on the given answer options. The answer to the question must be selected from the given answer options. Answer Options: {answer options 1} Question: {question 1} Answer: {answer 1} Answer Options: {answer options 2} Question: {question 2} Answer: {answer 2} Answer Options: {answer options 3} Question: {question 3} Answer: {answer 3} Answer Options: {answer options}

Table 13: The example question-answer pairs generated from different models on Covid QA.

Types	Samples
Augment w/ Seed Data	**Context**: polymerase chain reaction testing, the time lag between hospitalization and reporting was longer for early cases compared with that of more recent cases. Among the seven locations reporting importation, the total volume of inbound passengers from China was m = 63.1 million per year in 2017 [9] , of which 100q = 2.1% were from Wuhan [10] , a home of n = 19.0 million people as the catchment population of Wuhan airport. **Generated Question**: What was the total volume of inbound passengers from China to the seven locations reporting importation in 2017? **Generated Answer**: 63.1 million per year
Self-Instruct	**Context**: polymerase chain reaction testing, the time lag between hospitalization and reporting was longer for early cases compared with that of more recent cases. Among the seven locations reporting importation, the total volume of inbound passengers from China was m = 63.1 million per year in 2017 [9] , of which 100q = 2.1% were from Wuhan [10] , a home of n = 19.0 million people as the catchment population of Wuhan airport. **Generated Question**: What is the total volume of inbound passengers from China? **Generated Answer**: m = 63.1 million per year in 2017
RADA (Ours)	**Context**: MYTH: The flu is just a bad cold. Influenza may cause bad cold symptoms, like sore throat, runny nose, sneezing, hoarseness, and cough. But in the United States alone, 36,000 people die and more than 200,000 are hospitalized each year because of the flu. During the 2017/18 flu season, flu activity has significantly increased throughout the majority of the country with the A(H3N2) viruses predominating so far, according to the Centers for Disease Control (CDC). **Generated Question**: What are the consequences of getting the flu? **Generated Answer**: 36,000 people die and more than 200,000 are hospitalized each year because of the flu.

Table 14: The example question-answer pairs generated from different models on Policy QA.

Types	Samples
Augment w/ Seed Data	**Context**: IP Addresses. We use your IP address to help diagnose problems with our server, and to administer our Web site. Your IP address is used to help identify you and, in the case of the Coffee Review Marketplace, your shopping cart and to gather broad demographic information in the interest of improving our site and service. **Generated Question**: How does the company use IP addresses? **Generated Answer**: to help diagnose problems with our server, and to administer our Web site.
Self-Instruct	**Context**: IP Addresses. We use your IP address to help diagnose problems with our server, and to administer our Web site. Your IP address is used to help identify you and, in the case of the Coffee Review Marketplace, your shopping cart and to gather broad demographic information in the interest of improving our site and service. **Generated Question**: What information does the company use IP addresses for? **Generated Answer**: to help diagnose problems with our server, and to administer our Web site. Your IP address is used to help identify you and, in the case of the Coffee Review Marketplace, your shopping cart and to gather broad demographic information in the interest of improving our site and service.
RADA (Ours)	**Context**: The 1998 Data Protection Act sets the rules for how UK businesses can gather, store and use people confidential data. The Act is organised around eight principles of data protection. They include the idea that personal data should be obtained and processed fairly and lawfully, that only the right data should be used, that data should be accurate and so on. Principle Seven covers data security. It states: **Generated Question**: What is the main purpose of the 1998 Data Protection Act? **Generated Answer**: The 1998 Data Protection Act sets the rules for how UK businesses can gather, store and use people confidential data.

Table 15: The example question-answer pairs generated from different models on Tech QA.

Types	Samples
Augment w/ Seed Data	**Context**: VULNERABILITY DETAILS CVEID: CVE-2015-1920 [http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-1920] DESCRIPTION: WebSphere Application Server could allow a remote attacker to execute arbitrary code by connecting to a management port and executing a specific sequence of instructions. CVSS Base Score: 9.3 CVSS Temporal Score: See https://exchange.xforce.ibmcloud.com/vulnerabilities/102404 [https://exchange.xforce.ibmcloud.com/vulnerabilities/102404] for the current score CVSS Environmental Score*: Undefined CVSS Vector: (AV:N/AC:M/Au:N/C:C/I:C/A:C) **Generated Question**: What is the CVSS Base Score of the vulnerability in IBM WebSphere Application Server? **Generated Answer**: The CVSS Base Score of the vulnerability in IBM WebSphere Application Server is 9.3.
Self-Instruct	**Context**: VULNERABILITY DETAILS CVEID: CVE-2015-1920 [http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-1920] DESCRIPTION: WebSphere Application Server could allow a remote attacker to execute arbitrary code by connecting to a management port and executing a specific sequence of instructions. CVSS Base Score: 9.3 CVSS Temporal Score: See https://exchange.xforce.ibmcloud.com/vulnerabilities/102404 [https://exchange.xforce.ibmcloud.com/vulnerabilities/102404] for the current score CVSS Environmental Score*: Undefined CVSS Vector: (AV:N/AC:M/Au:N/C:C/I:C/A:C) **Generated Question**: What is the CVSS score for the vulnerability in WebSphere Application Server? **Generated Answer**: CVSS Base Score 9.3.
RADA (Ours)	**Context**: The Wii Homebrew Applications (such as The Homebrew Channel, Emulators, Games, Media players, etc.), USB Loader GX's NAND emulation feature and DIOS MIOS (Lite) require a FAT32 partition format. You can use a FAT32 SD/SDHC card for that purpose and set your USB hard drive as NTFS or Ext partition format. **Generated Question**: What partition format is required for certain Wii Homebrew applications? **Generated Answer**: FAT32