Task Oriented In-Domain Data Augmentation

Xiao Liang[1,3*] , **Xinyu Hu**[2*], **Simiao Zuo**[2], **Yeyun Gong**[3†] , **Qiang Lou**[2], **Yi Liu**[2],
Shao-Lun Huang[1], **Jian Jiao**[2†]
[1]Tsinghua University [2]Microsoft AI [3]Microsoft Research
`{liangx22@mails, shaolun.huang@sz}.tsinghua.edu.cn`
`{xinyuhu,simiaozuo,yegong,qilou,lewisliu,jian.jiao}@microsoft.com`

Abstract

Large Language Models (LLMs) have shown superior performance in various applications and fields. To achieve better performance on specialized domains such as law and advertisement, LLMs are often continue pre-trained on in-domain data. However, existing approaches suffer from two major issues. First, in-domain data are scarce compared with general domain-agnostic data. Second, data used for continual pre-training are not task-aware, such that they may not be helpful to downstream applications. We propose TRAIT, a task-oriented in-domain data augmentation framework. Our framework is divided into two parts: in-domain data selection and task-oriented synthetic passage generation. The data selection strategy identifies and selects a large amount of in-domain data from general corpora, and thus significantly enriches domain knowledge in the continual pre-training data. The synthetic passages contain guidance on how to use domain knowledge to answer questions about downstream tasks. By training on such passages, the model aligns with the need of downstream applications. We adapt LLMs to two domains: advertisement and math. On average, TRAIT improves LLM performance by 8% in the advertisement domain and 7.5% in the math domain.

1 Introduction

Large language models (LLMs) have achieved significant performance improvements in various applications such as language modeling (Brown et al., 2020; Touvron et al., 2023; Chowdhery et al., 2023) and visual understanding (Radford et al., 2021). They have also shown superior performance in fields such as finance (Xie et al., 2023b), e-commerce (Ma et al., 2023) and healthcare (Bakhshandeh, 2023). However, the models are usually trained on a large amount of general domain-agnostic data, such as web corpora. Because of the lack of domain-specific training, LLMs suffer from subpar performance when directly applied to certain domains such as advertisement.

To adapt LLMs to a specific domain, continual pre-training methods (Gururangan et al., 2020) are commonly applied. In particular, the LLM is continual pre-trained on in-domain corpora, such that it can acquire domain knowledge and better adapt to downstream tasks. Existing works (Cheng et al., 2023) have shown that such a technique drastically improves performance of LLMs on domains such as law and bio-medicine.

There are two major issues when continual pre-training LLMs. First, in-domain data are scarce. LLMs are pre-trained on large domain-agnostic corpora. For example, the web corpus used for pre-training contains more than ten trillion tokens. However, domain-specific data are magnitudes smaller, i.e., the ads in-domain corpus in our experiments contains only several billion tokens. Such a data scarcity issue significantly hinders model performance after continual pre-training.

Second, in-domain data used for continual pre-training are not task-oriented. Many existing works (Achiam et al., 2023; Li et al., 2023; Liu et al., 2024; Shao et al., 2024) focus on generating or selecting in-domain data without considering the downstream tasks. That is, the continual pre-training data are often passages that describe keywords/concepts of the target domain, which are generated or selected without considering whether the passages benefit downstream tasks.

We propose TRAIT (**T**ask o**R**iented in-dom**AI**n data augmen**T**ation), a data augmentation framework driven by downstream tasks of the domain. The framework is divided into two parts. First, to address the data scarcity issue of in-domain corpora, we propose a data selection strategy. The proposed algorithm identifies in-domain data from

---

[*] Equal contribution. Work done during Xiao Liang's internship at Microsoft Research.

[†] Corresponding authors.

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 20889–20907
November 12-16, 2024 ©2024 Association for Computational Linguistics

general corpora, and also applies a quality filter to ensure that the selected data have high educational value (Gunasekar et al., 2023). In practice, the amount of selected data is magnitudes larger than the in-domain dataset. For example, for the advertisement applications, the in-domain dataset contains about 1B tokens, and TRAIT selects an additional 15B tokens from web corpora.

Second, we propose a task-oriented synthetic passage generation guideline. Specifically, each generated passage contains multiple problems, where each problem comes from a different downstream task from the domain. Then, for each problem, TRAIT generates a problem-specific paragraph that suggests possible answers to the problem. Additionally, the synthetic passage also contains an *enlightenment paragraph*. This paragraph focuses on relationships among problems in the passage, including common and individual characteristics that are used to generate answers. Intuitively, the problem-specific paragraphs teach the model how to use techniques to solve a particular problem. And the enlightenment paragraph teaches the model common and unique aspects of all problems in the domain.

To fully exploit the power of TRAIT, we employ a two-stage training strategy for continual pre-training. In the first stage, the model is trained with in-domain data, including both the original in-domain corpora and the selected data. In this stage, the model adapts to the domain by learning domain knowledge. Then, in the second stage, the model is trained with the task-oriented passages. During this stage, the model learns how to use domain knowledge to solve problems, such that it better aligns with the need of downstream tasks.

We conduct extensive experiments by adapting LLMs to two domains: advertisement and math. For the advertisement domain we consider 7 downstream tasks, and TRAIT improves existing continual pre-training methods by 6.5% average score, while improving the base LLM (without continual pre-training) by 8%. For the math domain we consider 9 downstream tasks, where TRAIT outperforms the baseline by 5.5% average score and outperforms the base LLM by 7.5%. For the challenging MATH task in the math domain, TRAIT outperforms the base LLM by over 15%.

2 Method

2.1 Overview

We propose TRAIT, a data augmentation framework with two components. First, we propose a data selection strategy to select in-domain data from general corpus. In this way, we can significantly enlarge the domain-specific training data, such that the data contain more domain knowledge compared with the original small in-domain corpus. Second, we propose a guideline to generate task-oriented passages from in-domain data. The synthetic passages focus on using domain knowledge to solve the given tasks.

We use both the in-domain data and the synthetic passages to continual pre-train LLMs. Specifically, we first train the model on in-domain data, such that the model can learn more domain knowledge. Then, we train the model on the task-oriented synthetic passages. During this stage, the model learns to solve downstream tasks using the acquired domain knowledge. The proposed data augmentation framework and training strategy can drastically improve model performance by adapting LLMs to specific domains.

2.2 In-Domain Data Selection

In practice, the size of general corpus is orders of magnitude larger than domain-specific corpus. For example, the ads domain corpus contains about 1B tokens in our experiments, while the general web corpus contains trillions of tokens. To alleviate such a data scarcity issue, we propose to select in-domain data from general corpus.

We train a FastText (Joulin et al., 2017) classifier to identify in-domain data from large amount of domain agnostic data. Specifically, to train the FastText classifier, we select a certain number of in-domain data as positive samples and the same amount of out-of-domain data as negative samples. The trained binary classifier is then used to select in-domain data from the general corpus (e.g., the web corpus).

We apply a filter to ensure that the in-domain data (both the original in-domain corpus and the selected data) have high educational value (Gunasekar et al., 2023). In this way, we can boost the quality of the filtered in-domain data, which in turn improves performance of the models.

The proposed data selection strategy has two benefits. First, it can significantly enrich in-domain data. In practice, the amount of selected data is

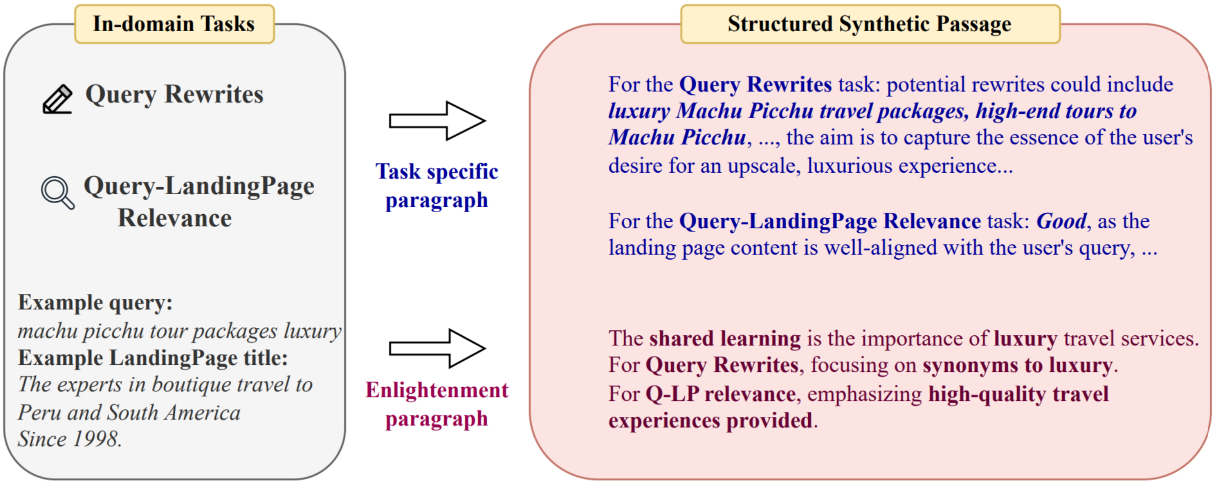

In-domain Tasks
Query Rewrites
Query-LandingPage Relevance
Example query:
machu picchu tour packages luxury
Example LandingPage title:
The experts in boutique travel to Peru and South America Since 1998.
Task specific paragraph
Enlightenment paragraph
Structured Synthetic Passage
For the Query Rewrites task: potential rewrites could include luxury Machu Picchu travel packages, high-end tours to Machu Picchu, ..., the aim is to capture the essence of the user's desire for an upscale, luxurious experience...
For the Query-LandingPage Relevance task: Good, as the landing page content is well-aligned with the user's query, ...
The shared learning is the importance of luxury travel services.
For Query Rewrites, focusing on synonyms to luxury.
For Q-LP relevance, emphasizing high-quality travel experiences provided.

Figure 1: An example of a task-oriented synthetic passage on the ads domain. Left: two downstream tasks (Query Rewriting and Query-LandingPage Relevance) and inputs. Right: the structure of the generated passage, including two problem-specific paragraphs and an *enlightenment* paragraph.

magnitudes larger than the in-domain dataset. For example, the original ads domain corpus contains about 1B tokens in our experiments, and we select an additional 15B tokens from the web corpus (after selection and filtering). Second, the data selection strategy enables *replay* (Ibrahim et al., 2024), such that generality of LLMs is largely kept after continual pre-training (see Table 5 for experiments). In more details, for a specific LLM, replay happens when the continual pre-training data contain a certain amount of pre-training data (e.g., the web corpus). It has been observed that replay is crucial to keep LLM's generality (e.g., instruction following) after training.

2.3 Synthetic Data Generation

To adapt LLMs to a specific domain, we first train the model on in-domain data, such that the model can acquire domain knowledge. Another key aspect crucial to model performance is the model's ability to use such knowledge to solve downstream tasks. To address this issue, we propose a guideline to generate task-oriented synthetic passages. In this way, the model is aware of downstream tasks during continual pre-training, and thus model performance can be significantly improved. In the next section, we describe how to generate the task-oriented passages in detail.

3 Task-Oriented Passage Generation

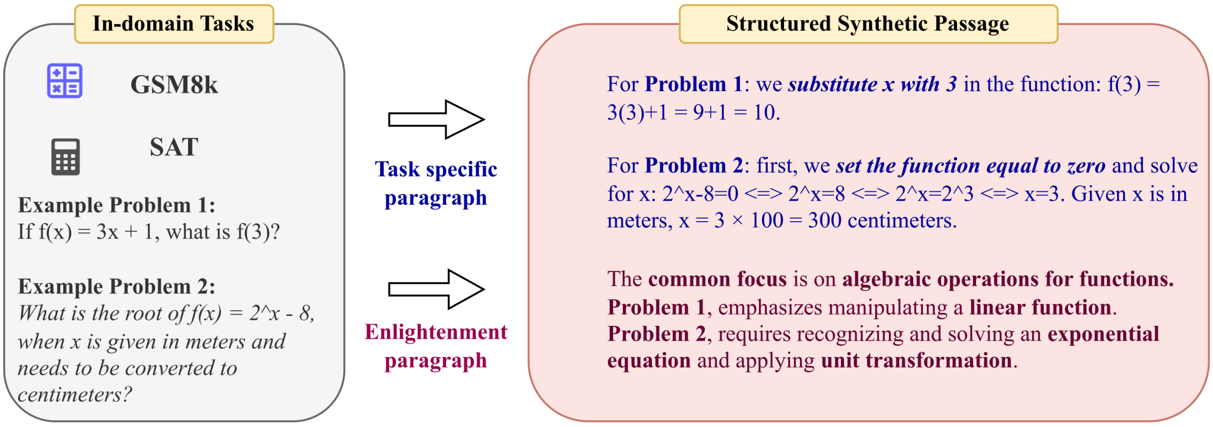
3.1 Guideline

The goal of the synthetic passages is to describe how to solve downstream tasks using domain knowledge. In TRAIT, each synthetic passage describes the common and individual characteristics of all downstream tasks in a domain. This resembles human learning: we learn how to solve individual problems, as well as common knowledge that can be applied to all problems.

We propose a guideline to generate task-oriented synthetic passages:

- ◇ We build each passage using several problems, where each problem comes from a different downstream task.
- ◇ Within a passage, for each problem we generate a problem-specific paragraph that suggests possible answers to the problem. Different prompts are used to generate paragraphs for different tasks, while the same prompt is used for problems from the same task.
- ◇ For each passage, we generate an *enlightenment paragraph*. This paragraph emphasizes relationships among problems, including shared and individual characteristics that are used to generate answers to the problems.

The enlightenment paragraph requires summarizing common and individual aspects of different problems from different downstream tasks. This is natural for certain domains. For example, in the ads domain in Figure 1, different questions in the passage ask about different aspects of the *same query*. As another example, in the finance domain, different features of the *same company* may be useful



Example Problem 1:
If f(x) = 3x + 1, what is f(3)?

Example Problem 2:
What is the root of f(x) = 2^x - 8, when x is given in meters and needs to be converted to centimeters?

For **Problem 1**: we ***substitute x with 3*** in the function: f(3) = 3(3)+1 = 9+1 = 10.

For **Problem 2**: first, we ***set the function equal to zero*** and solve for x: 2^x-8=0 <=> 2^x=8 <=> 2^x=2^3 <=> x=3. Given x is in meters, x = 3 × 100 = 300 centimeters.

The **common focus** is on **algebraic operations for functions.** **Problem 1**, emphasizes manipulating a **linear function**. **Problem 2**, requires recognizing and solving an **exponential equation** and applying **unit transformation**.

Figure 2: An example of a task-oriented synthetic passage on the math domain. Left: the selected two tasks (GSM8k and SAT) with an example problem from each task. Right: the structure of the generated passage, including two problem-specific paragraphs and an *enlightenment* paragraph.

for different tasks. We call these domains *entity-centered*. These domains focus on **understanding of entities** from various aspects.

On the other hand, in domains such as math and physics, the common aspects of different problems are not entities, but knowledge or techniques that can be applied to solve the problems. For example, in the math domain, each passage may contain multiple questions that require different techniques to solve, e.g., the GSM8k benchmark focuses on simple arithmetic, while the MATH benchmark focuses on logical reasoning. We call these domains *knowledge-centered*. These domains focus on using **universal knowledge** to solve problems.

3.2 Example: Task-Oriented Passage for Entity-Centered Domains

Recall that each synthetic passage is divided into two parts: problem-specific paragraphs and an enlightenment paragraph. We use ads domain as an example to illustrate how the two components are generated.

We select problems from different downstream tasks in the ads domain. In Figure 1, the passage contains two tasks (or two problems): Query Rewriting and Query-LandingPage Relevance.

For Query Rewriting, the task is to generate variations that maintain the search intent but diversify the expression. For the query "*Machu Picchu tour packages luxury*", the generated problem-specific paragraph looks like: *Potential rewrites for the query could include “luxury Machu Picchu travel package”, “high-end tours to Machu Picch”.*

For Query-LandingPage Relevance, the task is to decide whether the content of the landing page (the webpage to which the user is directed after clicking on an ad) addresses the intent of the search query. For the query "*Machu Picchu tour packages luxury*" and the landing page "*The experts in boutique travel ...*", the generated problem-specific paragraph looks like: *The landing page details, such as the expertise of the travel specialists ... directly correspond to the user’s search for a luxurious Machu Picchu tour, demonstrating a strong relevance.*

We also generate an enlightenment paragraph that focuses on relationships among the downstream tasks. For example, in Figure 1, the enlightenment paragraph is: *The shared learning is the importance of the luxury and personalized aspects of the travel service. For Query Rewrites, focusing on synonyms related to luxury and high-end services. Ensuring high relevance in Query-Landing Page Relevance emphasizes the high-quality travel experiences to Machu Picchu.*

The enlightenment paragraph severs as a central tenant of intelligence. It demonstrates which aspects of the entity are needed in all downstream tasks, and which aspects are tailored for a specific task. Such an explicit signal significantly boosts model performance. For example, for some downstream tasks in the ads domain, adding the enlightenment paragraph brings a 3% performance gain (see Table 3 for details).

3.3 Example: Task-Oriented Passage for Knowledge-Centered Domains

Different from entity-centered domains such as ads and finance, in knowledge-centered domains such as math and physics, the focus is on applying universal knowledge to solve problems. We use math domain as an example to demonstrate how we generate the task-oriented passages.

In Figure 2, we select two problems from the GSM8k and the SAT benchmarks. Then, we generate problem-specific paragraphs to solve the problems. Similar to the ads domain, in the enlightenment paragraph we summarize the common and individual techniques that are used to solve the problems. In more details, the enlightenment paragraph states that the common knowledge used to solve the two problems is "*algebraic operation for functions*". Each task requires additional techniques: the GSM8k problem requires "*manipulating a linear function*, while the SAT problem requires "*solving exponential equation*".

We remark that in knowledge-centered domains, the model learns universal knowledge that can be applied to all problems, i.e., the techniques learned from one problem are applicable to other problems. Therefore, the purpose of the enlightenment paragraph is to communicate about the techniques required to solve problems. The paragraph explicitly points out the common techniques needed by all downstream tasks, such that the model gains better awareness of the importance of such techniques. The model also learns task-specific techniques as pointed out by the enlightenment paragraph.

4 Data Preparation

We apply our data augmentation method, TRAIT, to two domains: advertisement (ads) and math. In this section, we detail the process for in-domain data selection and synthetic passage generation. Refer to Appendix. E for examples and more details.

4.1 Task-Oriented Passage Generation

Following the guidelines for task-oriented passage generation, we select problems from each downstream task and use GPT-4 to generate the full passage (prompt details can be found in Appendix. E). This approach is adaptable to any new target domain, leveraging GPT-4's understanding of various domains and its ability to handle diverse tasks within them. By selecting relevant problems and utilizing our generation prompt, our method ensures effective application across multiple domains.

4.2 In-Domain Data Selection

Ads domain. We train a domain-specific FastText classifier for in-domain data selection, as detailed in Section 2.2. First, we randomly select 500k positive samples from the ads in-domain corpus. We also select 500k negative samples from Slimpajama (Soboleva et al., 2023), alpaca (Taori et al., 2023), OpenHermes-2.5 (Teknium, 2023) and Tulu-v2 (Ivison et al., 2023). Then, for model training, we set the model dimension to 256, learning rate to 0.1, the maximum word n-gram length to 3, the minimum word occurrence to 3, and the epoch to 3. Next, we apply the trained classifier to select samples with the highest scores from the fineweb (Penedo et al., 2023) corpus. Finally, we apply a set of custom-designed rule-based filters (details can be found in Sec. B) along with a quality filter to both in-domain data and retrieved samples to ensure that each sample has an educational value over 1.5 (Gunasekar et al., 2023), where in total we select 15B filtered tokens.

Math domain. The domain-specific FastText classifier is trained similar to that in the ads domain. For the positive samples, we sample 200k examples from open-source benchmarks (such as GSM8k and SAT) and 200k samples from OpenWebMath (Paster et al., 2023). The negative samples are constructed similar with the ads domain. Due to the scarcity of math-related content in the general corpus, we retrieved the math data from a combination of MathPile (Wang et al., 2023) and Proof-Pile-2 (Azerbayev et al., 2023), resulting in a collection of around 5.5 billion tokens.

5 Experiments

We evaluate TRAIT by adapting LLMs to the ads and math domains via continual pre-training. In all the experiments, we use Mistral-7B (Jiang et al., 2023) and TinyLlama-1.1B (Zhang et al., 2024) as the base model. We compare TRAIT with two data selection baselines: (1) *Random sampling*, which randomly selects samples from open-source general corpora; and (2) *DSIR* (Xie et al., 2023a), an importance sampling strategy for selecting in-domain data from general corpora, such that the selected data distibutionally similar with the in-domain data. To promote fair comparisons, all models (including the base Mistral-7B model, baseline models, and TRAIT) are trained on the same amount of data with the same computational budget. Details about the training process can be found in Appendix A.

Method	QAC	QLP	QR		AG	DG	TG	TR	Avg. △
	Auc		Den.	Div.	Win Rate (%)				
`Few-shot Results`									
Mistral-7B	**69.48**	59.54	–	–	–	–	–	–	–
Random	63.94	60.29	–	–	42.20	55.15	53.62	47.75	-1.54%
DSIR	60.03	60.27	–	–	**57.43**	50.18	50.88	51.95	+1.41%
TRAIT	65.18	**65.91**	–	–	51.55	**60.42**	**54.60**	**55.10**	**+7.97%**
`Fine-tuned Zero-shot Results`									
Mistral-7B	82.93	78.81	5.29	3.06	–	–	–	–	–
Random	83.44	78.83	5.45	3.29	46.02	**52.35**	50.22	48.55	+0.68%
DSIR	84.10	79.96	5.48	3.36	50.58	49.38	51.98	50.32	+2.60%
TRAIT	**84.40**	**80.71**	**5.57**	**3.36**	**50.15**	51.95	**52.98**	**54.43**	**+4.79%**

Table 1: Evaluation results of downstream tasks in the ads domain. Here, *Avg* △ is the average relative improvement over all evaluation metrics for all tasks. Best results are highlighted in **bold**.

Method	GSM8K	MATH†	SVAMP	ASDiv	MAWPS	TAB	MQA	MMLU STEM	SAT	Avg.
`Results on TinyLlama-1.1B`										
Base	2.8	3.2	10.9	18.0	20.2	12.5	14.8	16.3	21.9	13.4
Random	4.5	6.4	13.6	24.6	28.1	10.8	15.0	13.6	12.5	14.3
DSIR	9.5	9.6	14.4	28.4	31.4	10.0	17.6	11.2	28.1	17.8
TRAIT	**16.9**	**11.4**	**25.8**	**37.9**	**33.3**	**15.5**	**23.6**	**17.8**	**31.3**	**23.7**
`Results on Mistral-v0.1-7B`										
Base	40.9	12.4	65.4	68.5	87.4	52.7	34.6	49.3	65.6	53.0
Random	34.8	14.0	60.4	65.2	82.4	39.7	34.9	46.4	56.2	48.2
DSIR	46.4	22.4	64.5	72.7	88.0	47.1	38.6	43.2	71.9	55.0
TRAIT	**56.4**	**28.0**	**71.8**	**76.0**	**89.5**	**53.1**	**46.1**	**49.5**	**75.0**	**60.5**

Table 2: Few-shot CoT reasoning results of downstream tasks in the math domain. For MATH†, evaluation is performed on OpenAI's MATH subset (Lightman et al., 2023), as the original test samples may be included in public training sets. Best results are highlighted in **bold**.

5.1 Ads Domain

Downstream tasks. We consider seven tasks within the ads domain. There are two classification tasks: Query-AdCopy Relevance (**QAC**) examines the relevance of a user's query to the ad copy, while Query-LandingPage Relevance (**QLP**) assesses relevance between a user's query and the advertisement's landing page content. For generation tasks, we focus on generating dynamic content: Query Rewriting (**QR**) generates rewrites of user queries, Ad Copy Generation (**AG**) creates complete ad copy directly, and both Description Generation (**DG**) and Title Generation (**TG**) develop concise descriptions and titles from the ad's landing page information. Additionally, Title Rewriting (**TR**) enhances user engagement by refining the ad title to better suit the user's query and the original title context.

Evaluation settings. We evaluate TRAIT under both few-shot and fine-tuning settings. Each task contains 5k test samples. For the fine-tuning setting, each task contains 30k training samples.

◇ For the two natural language understanding tasks (QAC and QLP), we adopt Area under curve (Auc) as the evaluation metric.

◇ For AG, DG, TG and TR, we use ChatGPT (OpenAI, 2022) to calculate the winning rate of TRAIT compared with the Mistral-7B model. Specifically, we prompt ChatGPT to choose the better answer from responses generated by our model and Mistral-7B. In order to mitigate ChatGPT's positional bias for evaluation (Chen et al., 2024), we swap the positions of the two responses and prompt ChatGPT again to choose the better answer. We average the outcomes from the two rounds as the final winning rate.

◇ The evaluation metrics for QR consist of diversity (Div.) and density (Den.), with details provided in Appendix D.

Results. Experimental results are summarized in Table 1. From the results, we see that TRAIT significantly outperforms both the Mistral-7B model and the baselines. Specifically, TRAIT achieves average increases of 8.0% and 4.8% across all downstream tasks compared with Mistral-7B in the few-shot and fine-tuning settings, respectively. And the proposed framework outperforms the best performing baseline by 6.5% and and 2.2% in the few-shot and fine-tuning settings, respectively. In the few-shot setting, TRAIT outperforms all the baselines in 4/6 tasks; while in the fine-tuning setting, the proposed framework performs the best in 6/7 tasks.

5.2 Math Domain

Downstream tasks. We evaluate the models across nine mainstream benchmarks: GSM8k (Cobbe et al., 2021), MATH (Hendrycks et al., 2021), SVAMP (Patel et al., 2021), ASDIV (Miao et al., 2021), MAWPS (Koncel-Kedziorski et al., 2016), TabMWP (TAB) (Lu et al., 2022), MathQA (MQA) (Amini et al., 2019), MMLU-STEM (Hendrycks et al., 2020), and SAT (Azerbayev et al., 2023). For evaluation, we adopt the math evaluation suite[1].

Evaluation settings. For all tasks, we evaluate under the few-shot chain-of-thought (CoT) (Wei et al., 2022) setting. We use accuracy as the final evaluation metric.

Results. As shown in Table 2, our continual pretrained models achieve absolute average accuracy improvements of 7.5% and 10.3% across all benchmarks compared with Mistral-7B and TinyLlama-1.1B base models, with significant gains of 15.6% and 8.2% on the most challenging MATH benchmark, respectively. We remark that TRAIT outperforms all the baselines in all the tasks.

5.3 Analysis

Effectiveness of TRAIT. In Figure 3, we see that the general data is far from the original in-domain data, indicating the necessity for domain-adaptive continual pretraining. The downstream tasks are distributed in various clusters, in proximity to the in-domain data, but not fully covered by it. For TRAIT, the mix of selected in-domain data and synthetic passages perfectly aligns with the downstream tasks, reflecting the task-awareness nature of our approach.

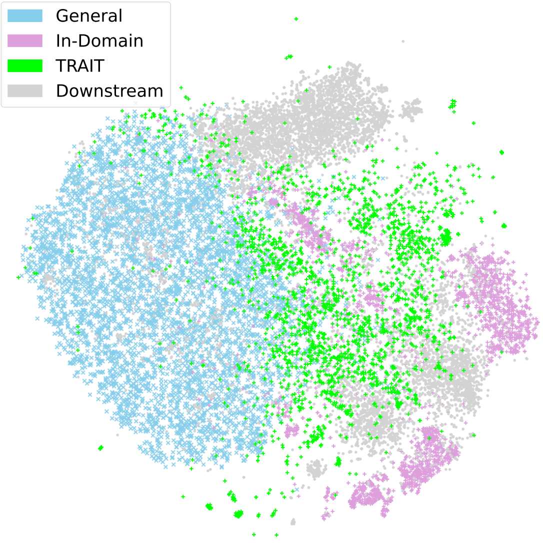


Figure 3: Visualization of samples from the general corpus, the original in-domain ads corpus, ads downstream tasks, and TRAIT (including both selected in-domain data and synthetic passages). We use Spacy (Honnibal and Montani, 2017) for embedding and t-SNE (Van der Maaten and Hinton, 2008) for visualization.

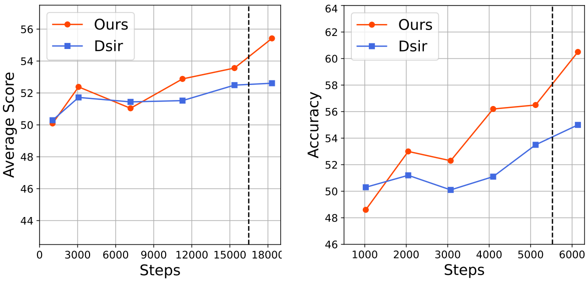


Figure 4: Left: The average winning rate of 4 ads generation tasks (AG, DG, TG and TR) during continual pre-training. Right: The average few-shot accuracy of all math tasks during continual pre-training.

In Table 4, we see the effect of each data component. The second row confirms the benefit of the original in-domain data, showing an average 1% performance gain across all tasks. A more notable contribution comes from TRAIT augmented data, with a nearly 5% gain observed, showing effectiveness of our data augmentation strategy.

Moreover, the benefit of the enlightenment paragraph is significant, as shown in Table 3. It reinforces a deeper understanding of queries in the ads domain and focuses on shared problem-solving techniques in the math domain.

Two-stage vs. Single-stage training. The performance of downstream tasks during continual pre-training is documented in Figure 4. In the first stage, where the aim is to learn new in-domain knowledge, we observe fluctuations in downstream performance as new knowledge, which may not be directly relevant to the tasks, is acquired. In the sec-

---

[1] math-evaluation-harness

	Ads Domain					Math Domain			
	QAC	**QLP**	**DG**	**TG**	**TR**	**GSM8k**	**SAT**	**MAWPS**	**MATH**†
TRAIT	84.40	80.71	51.95	52.98	54.43	56.4	75.0	89.5	28.0
w/o E.P.	83.77	79.80	50.95	51.92	51.58	54.6	68.8	89.7	27.2
w/o two-stage	83.23	79.84	50.22	51.50	51.32	53.2	59.4	89.3	25.2

Table 3: Effectiveness of the enlightenment paragraph and the two-stage training approach. We adopt the fine-tuning settings for the ads domain and the few-shot settings for the math domain. Here *w/o E.P.* means the model is trained without the enlightenment paragraphs.

In-D.	**Sel.**	**Syn.**	**QAC**	**QLP**	**TG**	**TR**
Mistral-7B			82.93	78.81	–	–
✓			82.50	79.18	50.73	50.38
	✓		82.47	79.35	52.05	48.05
✓		✓	82.64	79.17	52.88	48.20
	✓	✓	83.33	80.21	50.85	50.00
✓	✓	✓	**84.40**	**80.71**	**52.98**	**54.43**

Table 4: Performance of models continual pre-trained on different data. Models are evaluated on the ads domain under the fine-tuning setting. Here, *In-D.* means the original in-domain corpus, *Sel.* means selected in-domain data, and *Syn.* means synthetic passages.

	BBH	**ARC**	**HellaSwag**	**AgiEval**
Mistral-7B	55.91	70.55	61.25	32.64
In-Domain	49.84	65.24	59.57	30.43
TRAIT	53.06	67.59	60.34	32.26

Table 5: Few-shot evaluation of models trained on different data on general benchmarks. Here, *In-Domain* means the model is continual pre-trained on ads in-domain corpus (without selected data).

ond stage, the focus shifts to applying the learned knowledge to solve downstream tasks directly, resulting in a larger upward trend in task improvement. The overall benefit of adopting the two-stage training compared to a mixed single stage is significant, as shown in Table 3.

Generality after continual pre-training. Using the original in-domain data, the generality of LLMs deteriorates significantly, as shown in Table 5, with the model's performance on BBH decreasing from 55.91 to 49.84. In contrast, the model using TRAIT retains much of the generality. This is because, during the first stage, we train on the selected in-domain data from the general corpus as a knowledge replay (Ibrahim et al., 2024) and refocus on the target domain. Additionally, high-quality synthetic data ensures the model is trained with extensive reasoning.

6 Related Work

6.1 Data Augmentation for Language Models

Data selection is essential for the effective training of LLMs, as it significantly influences their performance. Common data selection methods include heuristic-based quality filters (Computer, 2023; Soldaini et al., 2024), lightweight classifiers (Joulin et al., 2017; Brown et al., 2020), and perplexity (PPL)-based models (Heafield, 2011; Wenzek et al., 2019), which are often developed using curated sources such as Wikipedia. For domain-specific data, techniques usually involve extracting information from the open web using heuristics or applying specialized classifiers to ensure relevance (Ma et al., 2023; Xie et al., 2023b). Other approaches select data based on their added value compared to typical examples from the target domain, or employ n-gram hash models to identify samples closely related to that domain (Axelrod, 2017; Feng et al., 2022; Xie et al., 2023a).

The use of synthetic data is becoming a key strategy for augmenting the training of LLMs, particularly useful in areas like mathematical reasoning (Gou et al., 2023; Huang et al., 2024; Toshniwal et al., 2024; Li et al., 2024a) and general instruction following (Wang et al., 2022; Xu et al., 2023; Li et al., 2024b). The Phi series highlights the effectiveness of models trained solely on "textbook quality" synthetic data (Gunasekar et al., 2023; Li et al., 2023; Abdin et al., 2024).

6.2 Continual Pre-Training of LLMs

Continual pre-training is increasingly recognized as an effective way to adapt large language models (LLMs) incrementally to new data or changes in domain focus without complete retraining. This method ensures the continuous integration of new knowledge, maintaining the model's relevance and effectiveness (Jin et al., 2021; Loureiro et al., 2022). Gururangan et al. (2020) has shown that continual pre-training can significantly improve model performance across various domains. LLMs like

EcomGPT and FinPythia demonstrate the application of continual pre-training in e-commerce and finance, using data from the open web and Common Crawl to stay functional and relevant (Ma et al., 2023; Xie et al., 2023b).

7 Conclusion

This paper presents TRAIT, a task-oriented in-domain data augmentation framework for continual pre-training of large language models. The framework is divided into two parts. First, we select in-domain data from general domain-agnostic corpora to augment the training set. The augmented in-domain training corpus contain rich domain knowledge. Second, we generate task-oriented synthetic passages. These passages contain guidance on how to apply domain knowledge to answer questions about downstream tasks. We conduct extensive experiments by adapting LLMs to the advertisement and math domains. Experimental results validate the effectiveness of the proposed framework. Specifically, on average, TRAIT improves the base LLM (without continual pre-training) by over 5% on both domains.

Limitations

In this work, we apply the proposed framework to the advertisement and math domains. The effectiveness of our framework in other domains, such as finance and medical, requires further investigation. Additionally, in this work, we only investigate the effectiveness of the proposed framework on Mistral-7B and TinyLlama-1.1B. We leave evaluating our framework on other LLMs as future works.

Acknowledgements

We extend our gratitude to Yinchuan Xu and Albert Wu for their insight during the initial scoping of the problem, and to Chengling Qiu, Jiayuan Huang, Luca Hwang, and Zihan Wang for their invaluable feedback throughout the iterations of this project.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *arXiv preprint arXiv:1905.13319*.

Amittai Axelrod. 2017. Cynical selection of language model training data. *arXiv preprint arXiv:1709.02279*.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen McAleer, Albert Q Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2023. Llemma: An open language model for mathematics. *arXiv preprint arXiv:2310.10631*.

Sadra Bakhshandeh. 2023. Benchmarking medical large language models. *Nature Reviews Bioengineering*, 1(8):543–543.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024. Humans or llms as the judge? a study on judgement biases. *arXiv preprint arXiv:2402.10669*.

Daixuan Cheng, Shaohan Huang, and Furu Wei. 2023. Adapting large language models via reading comprehension. *arXiv preprint arXiv:2309.09530*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Together Computer. 2023. Redpajama: an open dataset for training large language models.

Yukun Feng, Patrick Xia, Benjamin Van Durme, and João Sedoc. 2022. Automatic document selection for efficient encoder pretraining. *arXiv preprint arXiv:2210.10951*.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yujiu Yang, Minlie Huang, Nan Duan, Weizhu Chen, et al. 2023. Tora: A tool-integrated reasoning agent for mathematical problem solving. *arXiv preprint arXiv:2309.17452*.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. 2023. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of the sixth workshop on statistical machine translation*, pages 187–197.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Matthew Honnibal and Ines Montani. 2017. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 7(1):411–420.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, et al. 2024. Minicpm: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*.

Yiming Huang, Xiao Liu, Yeyun Gong, Zhibin Gou, Yelong Shen, Nan Duan, and Weizhu Chen. 2024. Key-point-driven data synthesis with its enhancement on mathematical reasoning. *arXiv preprint arXiv:2403.02333*.

Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats L Richter, Quentin Anthony, Timothée Lesort, Eugene Belilovsky, and Irina Rish. 2024. Simple and scalable strategies to continually pre-train large language models. *arXiv preprint arXiv:2403.08763*.

Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, et al. 2023. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *arXiv preprint arXiv:2311.10702*.

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*.

Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. 2021. Lifelong pretraining: Continually adapting language models to emerging corpora. *arXiv preprint arXiv:2110.08534*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431. Association for Computational Linguistics.

Rik Koncel-Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. 2016. Mawps: A math word problem repository. In *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 1152–1157.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626.

Chen Li, Weiqi Wang, Jingcheng Hu, Yixuan Wei, Nanning Zheng, Han Hu, Zheng Zhang, and Houwen Peng. 2024a. Common 7b language models already possess strong math capabilities. *arXiv preprint arXiv:2403.04706*.

Haoran Li, Qingxiu Dong, Zhengyang Tang, Chaojun Wang, Xingxing Zhang, Haoyang Huang, Shaohan Huang, Xiaolong Huang, Zeqiang Huang, Dongdong Zhang, et al. 2024b. Synthetic data (almost) from scratch: Generalized instruction tuning for language models. *arXiv preprint arXiv:2402.13064*.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, et al. 2024. Best practices and lessons learned on synthetic data for language models. *arXiv preprint arXiv:2404.07503*.

Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-Collados. 2022. Timelms: Diachronic language models from twitter. *arXiv preprint arXiv:2202.03829*.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. 2022. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. *arXiv preprint arXiv:2209.14610*.

Shirong Ma, Shen Huang, Shulin Huang, Xiaobin Wang, Yangning Li, Hai-Tao Zheng, Pengjun Xie, Fei Huang, and Yong Jiang. 2023. Ecomgpt-ct: Continual pre-training of e-commerce large language models with semi-structured data. *arXiv preprint arXiv:2312.15696*.

Shen-Yun Miao, Chao-Chun Liang, and Keh-Yih Su. 2021. A diverse corpus for evaluating and developing english math word problem solvers. *arXiv preprint arXiv:2106.15772*.

OpenAI. 2022. Chatgpt: Optimizing language models for dialogue. OpenAI Blog.

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. 2023. Openwebmath: An open dataset of high-quality mathematical web text. *Preprint*, arXiv:2310.06786.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? *arXiv preprint arXiv:2103.07191*.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: Outperforming curated corpora with web data, and web data only. *Preprint*, arXiv:2306.01116.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, YK Li, Y Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Daria Soboleva, Faisal Al-Khateeb, Robert Myers, Jacob R Steeves, Joel Hestness, and Nolan Dey. 2023. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. 2024. Dolma: An open corpus of three trillion tokens for language model pretraining research. *arXiv preprint arXiv:2402.00159*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Teknium. 2023. Openhermes 2.5: An open dataset of synthetic data for generalist llm assistants.

Shubham Toshniwal, Ivan Moshkov, Sean Narenthiran, Daria Gitman, Fei Jia, and Igor Gitman. 2024. Openmathinstruct-1: A 1.8 million math instruction tuning dataset. *arXiv preprint arXiv: Arxiv-2402.10176*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*.

Zengzhi Wang, Rui Xia, and Pengfei Liu. 2023. Generative ai for math: Part i – mathpile: A billion-token-scale pretraining corpus for math. *arXiv preprint arXiv:2312.17120*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2019. Ccnet: Extracting high quality monolingual datasets from web crawl data. *arXiv preprint arXiv:1911.00359*.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. 2023a. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36:34201–34227.

Yong Xie, Karan Aggarwal, and Aitzaz Ahmad. 2023b. Efficient continual pre-training for building domain specific large language models. *arXiv preprint arXiv:2311.08545*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*.

A Training Details

In this study, we select Mistral-7B (Jiang et al., 2023) as our base model for domain-adaptive continual pre-training. We perform continual pre-training using DeepSpeed (Rasley et al., 2020) with its ZeRO stage-1 optimizer, setting the batch size to 1M tokens and utilizing bf16 precision. We adopt a Warmup-Stable-Decay (WSD) learning rate scheduler (Hu et al., 2024) with a maximum learning rate of 2e-5, involving a 3% warm-up phase and an exponential decay phase during the last 10% of the training process.

During fine-tuning, we train all the models on downstream tasks for 5 epochs, using a cosine learning rate schedule with a maximum rate of 5e-6 and a 3% warm-up phase. For both few-shot and fine-tuned settings, we use vLLM (0.4.2) (Kwon et al., 2023) to accelerate inference.

B Filters for Pre-training Corpora

For the in-domain corpus in ads domain, after extracting text from webpages, we filter them through the following four processes:

1. Filtering search ad samples containing personally identifiable information (PII) using a rule-based filter (e.g., checking for email addresses or phone numbers).
2. Filtering out non-English samples and those with low English classifier scores using a fastText-based language identifier. These latter samples often contain a large number of meaningless symbols.
3. Filtering samples that are either too long or too short, as these samples often contain noisy texts.
4. Filtering samples where all words start with a certain symbol (e.g., +, #) using a Regular Expression-based refiner.

These rule-based filters are defined by observing the initial retrieved ad samples and should be adapted according to the data from different domains. Note that we do not further filter the retrieved math samples, which originate from the MathPile (Wang et al., 2023) and Proof-Pile-2 (Azerbayev et al., 2023) corpora and have already been strictly filtered.

C Side Effects by In-domain Corpus

Due to the helpfulness of the in-domain raw corpus, we do not stringent filtering by quality or diversity which will make its size significantly reduced. However, their quality may be notably inferior to the carefully curated pre-training corpus for open-source LLMs such as Mistral-7B (Jiang et al., 2023). Therefore, training on them could potentially impair the model's pre-trained general capabilities, such as its instruction-following ability. As shown in Figure 5, when training the model solely on our in-domain corpus and evaluating its few-shot performance on downstream generation tasks, the model fails to provide accurate responses, generating only repetitive and meaningless words, while the base model successfully responds to prompts. Thanks to the integration of high-quality retrieved and synthetic data, the model maintains its pre-trained capabilities during continual training on our curated corpus, which aligns with the conclusion in (Ibrahim et al., 2024) that a replay during continual training could preserve the model's pre-trained abilities.

D Evaluation Details of Query Rewriting

We generate ten query rewrites for each query and apply a quality filter to score the rewrites as either *Good* or *Bad*, based on their alignment with the original user query's intent. Density (Den) is defined as the total average number of good query rewrites per query. For diversity (Div), we use an off-the-shelf clustering algorithm to determine the average number of clusters per query based on all query rewrites. Both metrics range from 0 to 10.

E Examples of Prompts and Training Data

In this section, we demonstrate examples of prompts we used along with all unlabeled and downstream fine-tuning data. Specifically, we present an example of the prompt for synthetic data in Figure 6, an in-domain ad sample in Figure 7, a synthetic ad sample in Figure 8, a synthetic math sample in Figure 9, retrieved ad and math samples in Figure 10, and the downstream fine-tuning data for the ads domain in Figure 11 and Figure 12.

### **Prompt**:
You are an expert in advertisement and your task is to generate a creative ad given its content.
Here is the content of the advertisement:
DocumentTitle: Flow Meters & Controllers | Ferntree Gully, VIC | Anri § Paragraph: Florite manufacture a range of high quality microcomputer- based controllers suitable for any commercial, industrial or research application such as: ; Flow Control – Power supply/readout/control units for Mass Flow Meters (MFM's)MFC' programmable measurement units. Panel mount or bench top. Also suitable for Pressure, temperature, level & any process control. call our team on 03 9752 4057. § OtherPrimaryContent: en/ mfd ;mcmflow. § VisualTitle: Reliable Flow/ Controllers for Any Industry § MetaDescription: Controls supply flow meters & controllers for businesses Australia-wide. Call 03 9752 4057 or click here to learn more. §
The generated creative ad is:

### **Base Model**:
Flow Control: The Ultimate in Flow Metering. Experience the Elegance.

### **Model Continual Trained on In-domain Unlabeled Data**:
5' x 2' 3' x 8' 4' 6' x 10' x 12' 7' x 16' 9' x 13' x 14' x 15' x 18' x 20' x 23' x 24' x 27' x 29' x 30' x 40' x 50' 11' x 26' x 28' x 32' x 36' x 44' 17' x 60' 22' x 45' x 70' x 75' x 80' x 90' x 105' x 110' x 120' x 130' x 100' x 140' x 160' x 180' x 200' x 220' x 240' x 270' x 300' x 190' x 250' x 280' x 260' x 310'

### **Model Trained on Our Curated Corpus**:
100% Same-Day Delivery. Gophr: The Future of Last Mile Delivery.

Figure 5: An example of model degradation: the model, when continually pretrained on the original in-domain corpus, exhibits repetitive and nonsensical text generation, unlike its performance with the base model and TRAIT corpus, where this issue is absent.

Prompt for Generating Task-Oriented Synthetic Data

#### Structured Guideline for Passage Generation

#### Inputs Required:
- **Questions**: The question for each task.

#### Passage Generation Steps:
- Task specific: For each of the downstream tasks listed below, write one paragraph analyzing the potential answers and the reasoning process associated with each. Please list the answer explicitly.
- Enlightenment: After writing paragraphs for all tasks, highlighting shared learnings across all tasks and distinct problem solving tricks for each task, specifically the current problem.

#### Quality Considerations:
- Ensure coherence and logical flow throughout the passage.
- Maintain a concise and clear writing style, avoiding redundancy and focusing on summarizing key points.

#### Input: Please return only the generated passage between tags <Passage></Passage> given below input.
- {Task 1}: {Problem 1}
- {Task 2}: {Problem 2}
- {Task 3}: {Problem 3}
- {Task 4}: {Problem 4}

Figure 6: The prompt utilized for generating the task-oriented synthetic data.

Example for an unlabeled ad sample

Advertisement Title: Vision Reading Glasse

Advertisement Description: Save On Vision Reading Glasses. Everyday Low Prices!

Key words: hand held lighted magnifying glass

Document Title: Hand Held Lighted Magnifying Glass - Walmart.com

Heading: Results for "Hand Held Lighted Magnifying Glass" (1000+) Options

Primary Content: Departments Brand Speed Subscription Availability Special Offers Customer Rating Features Magnification Color Material Manufacturer Part Number Count Retailer Gifting Price when purchased online Best seller Sponsored Magnifying Glass with 18 LED Light, Meromore 30x Handheld Magnifier for Reading Save with Shipping, arrives in 2 days Meromore Magnifying Glass, Lighted Magnifying Glass with 3 LED, Handheld Magnifier with 3x 45x Magnification for Kids Reading Meromore 30x Magnifying Glass with 18 LED Lights, Black MagniPros 3 Ultra Bright LED Lights 3X 4.5X 25X Power Handheld Reading Magnifying Glass with Light- Ideal for Reading Small Prints, Map, Coins, Inspection and Jewelry Loupe Free shipping, arrives in 3+ days Magnifying Glass with Light, Lighted Magnifying Glass, 5X Handheld Pocket Magnifier Small Illuminated Folding Hand Held Lighted Magnifier for Reading Coins Hobby Travel - 45 Mm Diameter 2 options KEKOY Handheld Magnifying Glass with Light and Double Lens, Ultra Clear USB Charging Magnifying Glass for Close Work with 6x 9x 15x Detachable Lens, Strong Maginifier for Reading RockDaMic Professional Magnifying Glass with Light (3X / 45x) Large Lighted Handheld Glass Magnifier Lupa for Reading, Jewelry, Coins, Stamps, Fine Print - Strongest Magnify for Kids & Seniors 2 sizes Large 4.35in Magnifying Glass 10X 35X with Light and Stand, Handheld Standing LED Illuminated Magnifier, Folding Reading Magnifying Glass with for Seniors Read, Cross Stitch, Map, Jewelry Meromore Magnifying Glass.

Figure 7: An example of an unlabeled in-domain sample from the ads domain.

Example for Syhthetic Ads Data

For the Query Rewrites (QR) task, the potential answers revolve around creating variations of the user query "machu picchu tour packages luxury." Possible rewrites could include "luxury Machu Picchu travel packages," "high-end Machu Picchu tours," and "premium tour packages to Machu Picchu." Each rewrite aims to capture the essence of the user's request while incorporating keywords that might improve the ad's visibility and relevance. The reasoning process involves ensuring that the rewrites maintain the focus on luxury and Machu Picchu, aligning with the user's intent and the ad's offer.

In the Ad Copy Generation (AG) task, the goal is to create a complete ad copy directly from the landing page content. This involves synthesizing the key elements of the landing page into a compelling ad. For example: "Discover High-End Machu Picchu Tours with Inca Expert. Since 1998, we've crafted bespoke travel experiences with the guidance of our award-winning specialists, including Emmy-winning filmmaker Kim MacQuarrie and World's Top Chef Virgilio Martinez. Experience Peru with private guides and boutique services. Visit us at https://www.incaexpert.com." The reasoning here includes selecting the most impressive and relevant details from the landing page to attract potential customers while maintaining a concise and engaging format.

For the Title Rewriting (TR) task, the focus is on refining the ad title to better suit the user's needs. Given the user's query, a more targeted title could be "Luxury Machu Picchu Tours by Peru Experts" or "Top-Rated High-End Machu Picchu Travel." The aim is to enhance engagement by clearly communicating the luxury aspect and the expertise of the travel firm, making it more appealing and relevant to the user's search.

In the Query-Ad Copy Relevance (QAC) task, the relevance of the user's query to the ad copy must be examined. The user query "machu picchu tour packages luxury" directly aligns with the ad copy, which promotes "High-End Machu Picchu Tours" and highlights boutique service and private guided tours. The ad copy effectively addresses the user's desire for a luxury travel experience, ensuring high relevance. The reasoning involves matching keywords and themes from the query with those in the ad copy to ensure they resonate well.

The Query-Landing Page Relevance (QLP) task assesses the relevance between the user's query and the advertisement's landing page content. The landing page details Inca Expert's specialization in boutique, high-end travel to Peru, featuring notable specialists and personalized services, which strongly align with the user's search for luxury Machu Picchu tour packages. The primary content supports the luxury and personalized experience sought by the user. The reasoning here involves confirming that the landing page provides substantial and relevant information that fulfills the user's query.

In conclusion, the key points across the tasks highlight the importance of aligning ad copy and titles with the user's intent for luxury travel, ensuring high relevance and engagement. Each task focuses on different aspects: Query Rewrites (QR) focusing on synonyms to luxury; Ad Copy Generation (AG) synthesizes key landing page information like personalized services provided by award-winning specialists in flim-making and cooking, into compelling ads; and Title Rewriting (TR) enhances engagement by clearly communicating the luxury aspect and the expertise of the travel firm. Query-Ad Copy Relevance (QAC) and Query-Landing Page Relevance (QLP) ensure consistency and alignment between user expectations, ad content, and landing page details for the high-quality travel experiences provided. These tasks collectively improve the user's experience by ensuring coherence and relevance across all touchpoints, ultimately driving engagement and satisfaction.

Figure 8: An example of generated synthetic data in ads domain.

Example for Syhthetic Math Data

###Task 1. For the below question: In the undersea world of Mathlantica, Aquatic Mathematicians communicate through positive integer pairs (a, b) where b<a<150. However, they can only understand the messages if both a/b and (a+2)/(b+2) are integers. How many such communicable pairs exist? As an additional challenge, they ask you, a surface-dweller, to add up the numbers 212, 321 and 123, which they inform you are in base 10, and provide the sum in base 10. Can you solve these underwater conundrums?,Enclose the final answer using \boxed{}. The plausible answer and explantion could be: For the first part, since $b|a$, $b+2|a+2$, then $\gcd(b, a) = b$ and $\gcd(b+2, a+2) = b+2$. By the Euclidean algorithm, these can be rewritten respectively as $\gcd(b, a-b) = b$ and $\gcd(b+2, a-b) = b+2$, which implies that both $b, b+2|a-b$. Also, as $\gcd(b, b+2) = 1$, it follows that $b(b+2)|a-b$. [1],Thus, for a given value of b, we need the number of multiples of $b(b+2)$ from 0 to $150-b$. It follows that there are $\left\lfloor \frac{150-b}{b(b+2)} \right\rfloor$ satisfactory positive integers for all integers $b \leq 150$. The answer is,

$$\sum_{b=1}^{148} \left\lfloor \frac{150-b}{b(b+2)} \right\rfloor = 74+24+12+6+4+3+2+1+1 = 127.$$

For the second part, adding 212_{10}, 321_{10} and 123_{10} together in base 10 gives us 656_{10}.,,So, the final answer is $127+656 = \boxed{783}$.,So the answer is $\boxed{783}$.

###Task 2. For the below question: In a space station, an astronaut is studying a parallelogram-shaped window named $WXYZ$. The angle WXY is four times the angle XYZ. Inside the window, there is a right circular truncated cone-shaped object. The smaller base of the object is on XY and the larger base is on WZ. The height of the object is equal to the length WY. The radius of the larger base is 10 cm and the radius of the smaller base is 5 cm. Can you help the astronaut calculate the volume of the object in cm^3?,Enclose the final answer using \boxed{}. The plausible answer and explantion could be: Firstly, we need to find the height of the truncated cone. Since the height is equal to the length WY, we need to find WY. To do this, we use the fact that the measure of angle WXY is 4 times the measure of angle XYZ. Because $\overline{WZ} \parallel \overline{XY}$, we have $\angle W + \angle X = 180^\circ$. Since $\angle W = 4\angle X$, we have $4\angle X + \angle X = 180^\circ$, so $5\angle X = 180^\circ$ and $\angle X = 36^\circ$. Therefore, $\angle W = 180^\circ - \angle X = 144^\circ$. Since $WXYZ$ is a parallelogram, $\angle WYZ = \angle X = 36^\circ$. Therefore, triangle WYZ is a right-angled triangle and $WY = 2WZ = 2*10 = 20$ cm.,,We "complete" the truncated cone by adding a smaller, similar cone atop the cut, forming a large cone. Since the small and large cone are similar, we have $x/5 = (x+20)/10$; solving yields $x = 20$. Hence the small cone has radius 5, height 20, and volume $(1/3)\pi(5^2)(20) = 500/3\pi$ and the large cone has radius 10, height 40, and volume $(1/3)\pi(10^2)(40) = 4000/3\pi$. The frustum's volume is the difference of these two volumes, or $4000/3\pi - 500/3\pi = \boxed{3500/3\pi}$ cubic cm.,So the answer is $\boxed{3500/3\pi}$.

###Task 3. For the below question: If $f(x) = 3x+1$, what is $f(3)$? The plausible answer and explantion could be: We have $f(3) = 3*3+1 = \boxed{10}$.

###Task 4. For the below question: In a music festival, there are certain speakers that play a specific song every 20 minutes. If a speaker played the song 5 minutes ago and will play it again in 30 minutes, how many such speakers are there that have been playing for less than 50 minutes? In the same festival, two bands are performing. The first band's performance progresses at a rate of 2 songs per minute, and the second band's performance progresses at a rate of 4 songs per minute. Both bands started performing from the same point (5,10) minutes ago. How far apart are the points where these bands finish their performances?,Enclose the final answer using \boxed{}. The plausible answer and explantion could be: There are two parts in this question.

Part 1: Adding -5 to both sides of $x+5 \equiv 30 \pmod{20}$ gives $x \equiv 30-5 \pmod{20}$. We find $30-5 = 25 \equiv 5 \pmod{20}$, so $x \equiv 5 \pmod{20}$. Thus the smallest positive integer that satisfies the given congruence is $x = 5$, and all the other solutions can be obtained by adding a multiple of 20 to 5. The next two such integers are 25 and 45. Since 45 is less than 50, there are $\boxed{3}$ integers less than 50 that satisfy the congruence $x+5 \equiv 30 \pmod{20}$.

Part 2: The two lines have equations

$$y - 10 = 2(x-5) \quad \text{and} \quad y - 10 = 4(x-5).$$

The x-intercepts, obtained by setting $y = 0$ in the respective equations, are 5 and 2.5. The distance between the points $(5, 0)$ and $(2.5, 0)$ is $\boxed{2.5}$.,So, for the first sub-question, the answer is $\boxed{3}$.,For the sub-question 2, the answer is $\boxed{2.5}$.

In analyzing the provided math tasks, we can identify several shared problem-solving techniques such as mathematical induction, unit transformation, and basic algorithmics. For instance, in Math task 1, the problem-solving involves a combination of number theory (specifically divisibility and the Euclidean algorithm) and arithmetic operations in different bases. Math task 2 requires geometric reasoning to find the dimensions of the shapes involved and then applying the formula for the volume of a truncated cone. Math task 3 is a straightforward application of function evaluation, while Math task 4 combines modular arithmetic with linear equations to solve the problem.

Each task also exhibits unique problem-solving techniques. Math task 1 uses number theory to find communicable pairs and base conversion for arithmetic operations. Math task 2 involves geometric properties of parallelograms and right-angled triangles, as well as similarity of shapes to calculate volume. Math task 3, being the simplest, only requires direct substitution in a linear function. Math task 4 uses modular arithmetic to determine the number of speakers and the concept of linear equations to find the distance between two points.

Figure 9: An example of generated synthetic data in math domain.

Example for Retrieved Ads

Acadeos is the best online learning sites In USA to achieve your professional goals through various USA e courses. Join our Online Academy US now. Learn US Abacus online through our online learning platforms in Acadeos. We use virtual Abacus USA for students to get better Abacus learning online USA.
Learn Alphabet Phonics USA through our online Online Education In United States in Acadeos. You can study phonics in your home by learning online. Improve your child's mathematical skill by employing your kid in Vedic maths online classes In USA. Join US Vedic Math Online Course In Acadeos now.
Get Dissertation Help Online In US from verified experts and buy a thesis paper In USA online with high quality in our Acadeos in an effective way. Improve your child's learning skills through story telling USA from our virtual learning academy in United States by joining your kids in Acadeos now.
Reach online science tutor In US and get best tutoring services United States. Connect with our experienced professionals in Acadeos to study online.
Access Skype math tutor US to get a best Online Mathematics Tutor In US for your kid to gain best mathematical knowledge from the best Math Tutoring USA.
Enroll your kid in our virtual schools USA to get virtual learning In US from our professionals In Acadeos to get online homework help for assignments.
Acadeos provides the best maths tutors online to help you with math homeworks. We offer a customised tuition plans and a student-centric approach.

Example for Retrieved Math

Exponential Growth Worksheet
• Page 1
1. If a quantity increases by the same percent r in each unit of time t, then the quantity is _____.
a. growing exponentially b. decreasing exponentially c. constant
### Solution:
If a quantity is increasing by the same percent r in each unit of time t, then the quantity is growing exponentially.

2. Which of the following equations represents exponential growth?
a. $y = r(1 + r)$ b. $y = r(1 + \mathrm{C})$ c. $y = \mathrm{C}r$ d. $y = \mathrm{C}(1 + r)t$
### Solution:
Exponential growth can be modeled by the equation y = C (1 + r)t, where C is the initial amount, r is the growth rate and t is the time.

3. The expression $(1 + r)$ is called ____ in the equation $y = \mathrm{C}(1 + r)t$.
a. decay factor b. growth factor c. decay and growth factors d. exponent
### Solution:
The expression (1 + r), in the equation y = C(1 + r)t is called growth factor.

4. The average length of a person's hair at birth is 0.36 inches. The length of the hair increases by about 10% each day during the first six weeks. Choose a model that represents the average length of the hair during the first six weeks.
a. $y = 0.36(1.1)t$ b. $y = -0.36(1.1)t$ c. $y = 1.1(0.36)t$ d. None of the above
### Solution:
Let y be the length of the hair during the first six weeks and t be the number of days.
y = C(1 + r)t [Write exponential growth model.]
= 0.36(1 + 0.10)t [Substitute C = 0.36 and r = 0.10.]
= 0.36(1.1)t
The model for the length of the hair in first six weeks is y = 0.36(1.1)t.

5. A bank pays 4% interest compounded yearly on a deposit of $900. What will be the balance in the account after 7 years?
a. $1288 b. $1088 c. $2376 d. $1188
### Solution: The exponential growth model is given by the equation, y = P(1 + r)t, where P is the initial amount, r is the growth rate and t is the number of years. = 900(1 + 0.04)7 Balance after 7 years [Substitute P = 900, t = 7 and r = 0.04.] = 900(1.04)7 = 900 x 1.32 = 1188 [Simplify.] The account balance after 7 years will be about$1188.

6. There are 20 bears in a zoo. What will be their population after 3 years, if the population doubles each year?
a. 160 bears b. 260 bears c. 60 bears d. 210 bears
### Solution:
The exponential growth model is given by the equation, y = C(1 + r)t, where C is the initial number, (1 + r) is the growth factor and t is the number of years.
Population after 3 years = 20(2)3 [Substitute C = 20, 1 + r = 2 and t = 3.]
= 160 [Simplify.]
There will be 160 bears after 3 years.

Figure 10: Examples of retrieved ads and Math unlabeled samples.

1. Example for Query-Ad Copy Relevance (QAC)

> > > **Prompt**:

You are an expert in advertisement and your task is to evaluate the relevance between a user input query and an advertisement.

Here are some attributes for the advertisment:

### begin advertisement

Actual Advertisement Title: Plumbers Near Me - Enter Your Zip Code To Start - View Quotes In Under 24 Hours

Actual Advertisement Description: Explore Professional Plumbers Who Specialize In Your Project Type. Get Up To 4 Estimates. Receive Accurate Quotes For Your Plumbing Project, So You Can Easily Save Time And Money.

### end advertisement

The user query is: kaufmann plumbing palm springs

Please evaluate that whether the advertisment is relavant to the user query. You can only answer with True or False.

The answer is (True or False):

> > > **Response**:

False

2. Example for Query-Landing Page Relevance (QLP)

> > > **Prompt**:

You are an expert in advertisement and your task is to evaluate the relevance between a user input query and an advertisement.

Here are some attributes for the advertisement:

### begin advertisement

Document Title: Verified Camping World Promo Code & Coupon Code August 2022

Visual Title: Camping World Promo Code & Coupon Code July 2022

Heading: Submit Coupon for Camping World Camping World Stats Camping World Top Coupon Codes and Offers Get Latest, Vitrified 30% Off Promo Code, Don't Pay Full Price! ADVERTISEMENT Save Your Money, Get 30% Off Coupon Code Big Deal Today, Up To 60% Offer Flash Sale! Up To 70% Off Coupon Code Up To 50% Off Today, Save Your Money Now! Camping World 10% Off Storewide Up To 46% On RV Sales Up To $500 Off Sleep Number Beds Up To 30% Off Stromberg RV Gear Take 15% Off Your Online Purchase Camping World Gain Up To 45% Off Awnings, Sunblockers & Replacement Fabrics Up To $250 Off Refrigerators, Washers & Dryers

Primary Content No Title No Heading: Continue to campingworld.com Rate 4.1 / 153 Votes With the advancement of technology, everybody began to pursue high-quality development, and coupomuscode.com is here to assist everyone in achieving this objective in a more comfortable manner. For your convenience, coupomuscode.com has carefully picked Camping World and coupons. The perfect bargain is when you can acquire not just the greatest goods, but also the greatest price, allowing you to use the money you save into something else. Check out the best Camping World coupons and save up to 80%. The best choice of today is Get Latest, Vitrified 30% Off Promo Code, Don't Pay Full Price!, don't miss it. Total Promo Codes Deals Max Discount Last Updated July 2022 Get Latest, Vitrified 30% Off Promo Code, Don't Pay Full Price! Save Your Money, Get 30% Off Coupon Code

### end advertisement

The user query is: camping world

Please evaluate that whether the advertisement is relavant to the user query. You can only answer with True or False.

The answer is (True or False):

> > > **Response**:

True

3. Example for Query Rewriting (QR)

> > > **Prompt**:

You are an expert in advertisement.

Given a specific search query A, generate 10 potential query B's that users might use to express the same intent as query A.

The goal is to craft diverse yet relevant variations of query B that could match the original intent of query A.

Focus on avoiding repetitions and ensuring that each variation of Query B is a distinctly different way of expressing the intent of Query A.

Output in a valid python list format and don't generate any other text. Now perform the tasks for the following input.

input: Query A: redfall release xbox

output: Query B:

> > > **Response**:

["when is redfall coming out on xbox", "redfall xbox launch date", "xbox redfall game release", "redfall availability on xbox", "how to get redfall on xbox", "redfall xbox debut", "xbox release of redfall game", "redfall game for xbox release time", "redfall on xbox launch details", "redfall xbox edition release info"]

Figure 11: Examples of the prompt and the labeled responses for the QAC, QLP, and QR tasks.

4. Example for Ad Copy Generation (AG)

> > > **Prompt**:
You are an expert in advertisement and your task is to craft the ad copy for an advertisement based on its content. The ad copy should provide a concise description of the advertisement that entices users before they click into the ad page.
Here is the content of the advertisement:
DocumentTitle: ENCINITAS Florist | Flower Delivery by ENCINITAS FLORIST § Heading: Flower Delivery in ENCINITAS, CA ; Browse Designs See all flowers ; Hawaiian Hugs and Kisses ; Always On My Mind™ Flower Bouquet ; Victorian Romance ; Over the Moon Bouquet ; The FTD Always Adored Bouquet ; Coastal Breeze ; Sun-Sational Sunflowersâ ; Calla lillies ; The Stunning Beauty™ Bouquet by FTD - VASE INCLUDED ; Teleflora's Desert Sunrise Bouquet ; New Sensations ; TULIPS SPECIAL ; ORCHIDS AND SUCCULENT Rustic ; Vintage Garden ; All White Arce ; Fields of Europe for Summer ; § OtherPrimaryContent: Next-day Delivery § MetaDescription: Best ENCINITAS florist. ENCINITAS FLORIST offers fresh flower delivery ENCINITAS. Save money by sending flowers directly with a Local Florist. § FirstGood: For an accessible ADA experience, please visit our Marketplace. Visit our Accessibility page for further assistance or to learn more about our efforts.
The generated ad copy is:
> > > **Response**:
Buy Flowers Online -TitleSep- Local Flower Source -Desc- Flower Shop in Encinitas Say it With Flowers!

5. Example for Ad Description Generation (DG)

> > > **Prompt**:
You are an expert in advertisement and your task is to generate a creative ad given its content.
Here is the content of the advertisement:
DocumentTitle: Fertilizer Granulator - Organic and NPK fertilizer making machines § Heading: Cat Litter Disc Fertilizer Granulator ; New Type Organic Fertilizer Granulator ; Organic Fertilizer Pin Granulator ; Rotary Drum Granulator ; Double roller extrusion granulator § OtherPrimaryContent: Home > Fertilizer Machines > Fertilizer Granulator ; Check details § FirstGood: Shunxin News. Shipping disc granulator fertilizer production line to America. BB and NPK fertilizer production lines shipping to Nigeria. Deliver 10 t/h granular fertilizer produc §
The generated creative ad is:
> > > **Response**:
From Trash to Treasure: Granulator Machine for Fertilizers

6. Example for Title Generation (TG)

> > > **Prompt**:
You are an expert in advertisement and your task is to craft the title for an advertisement based on its content. The ad copy should provide a concise description of the advertisement that entices users before they click into the ad page.
Here is the content of the advertisement:
DocumentTitle: Four Season Rain Gutters (1145223) § Paragraph: Four Seasons Rain Gutters has provided courteous, reliable service and high-quality gutters for customers in all of San Diego County, with an excellent history of customer satisfaction. We use only the best aluminum, copper, steel gutters, no matter how big or small the job is,ve got the experience to get the job done. Also, we are dedicated to providing our customers with quality workmanship, professionalism, reliability, punctuality, clean work, competitive pricing. Call us. § Heading: Business Information ; Hours of Operation § OtherPrimaryContent: Call us today at ; Escondido, CA 92027 ; Get Directions ; Phone ; Website ; https:www.fourseasonsraingutters.com/ §
The generated title is:
> > > **Response**:
High Quality Gutters Available

7. Example for Title Rewriting (TR)

> > > **Prompt**:
You are an expert in advertisement and your task is to rewrite compelling titles that resonates with user's query and enticits them to click on the ad, given the orginal titles and the user query.
The user query is: best medical supply stores near me
The original advertisement titles are: Home Health, Hospitals and More - Hopkins Medical Products - Hopkins Medical Supplies
The output rewritten advertisement titles are:
> > > **Response**:
Home Health - Hospital Supplies Near Me - Hopkins Medical Products

Figure 12: Examples of the prompt and the labeled responses for the AG, DG, TG and TR tasks.