Have Seen Me Before? Automating Dataset Updates Towards Reliable and Timely Evaluation

Anonymous ACL submission

Abstract

Due to the expanding capabilities and pre-training data, Large Language Models (LLMs) are facing increasingly serious evaluation challenges. On one hand, the data leakage issue cause over-estimation on existing benchmarks. On the other hand, periodically curating datasets manually is costly. In this paper, we propose to automate dataset updates for reliable and timely evaluation. The basic idea is to generate unseen and high-quality testing samples based on existing ones to mitigate leakage issues. In specific, we propose two strategies with systematically verification. First, the mimicking strategy employs LLMs to create new samples resembling existing ones, to the maximum extent preserving the stylistic of the original dataset. Our experiments demonstrate its evaluation stability across multiple instantiations and its effectiveness in dealing with data leakage issues in most cases. Second, for the cases that mimicking dataset works poorly, we design an extending strategy that adjusts the difficulty of the generated samples according to varying cognitive levels. This not only makes our evaluation more systematic, but also, with a balanced difficulty, even discern model capabilities better at fine-grained levels.

1 Instruction

Large Language Models (LLMs) have become essential tools in the field of Natural Language Processing, marking significant advancements as highlighted by a series of influential studies (Brown et al., 2020; Wei et al., 2022c; Mirowski et al., 2023). Trained on wide-ranging datasets, these models exhibit in-context learning, instruction following, and step-by-step reasoning capabilities (Wei et al., 2022b; Zhao et al., 2023). As the pre-trained data and capabilities of LLMs expand, how to curate suitable dataset for evaluation is becoming more and more challenging. On one hand, as highlighted by Li (2023); Zhou et al. (2023); Wei et al. (2023); Li and Flanigan (2023); Jacovi et al. (2023), data leakage or contamination, the phenomenon that examples from the evaluation set are also found in the training set (Li, 2023), is more common than ever. A straightforward solution, on the other hand, is to manually update the dataset at regular intervals, which is costly.

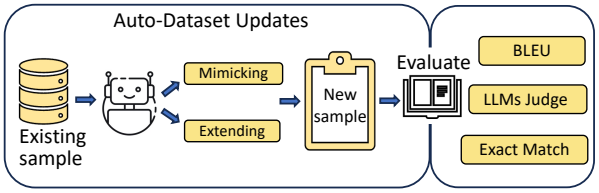

Figure 1: Auto-dataset update framework, where we deploy two strategies: mimicking and extending to update

In this paper, we aim to automate dataset updates for evaluation reliability and timeliness. When facing data leakage issues, we can simply replace with a new one without costly human efforts. Compared with building from scratch, our proposal is based on existing benchmarks, offering a more practical solution with minimal human effort. However, this is non-trival due to the following research questions.

- Can the updated datasets avoid the data leakage or contamination issue?
- Is it stable if we dynamically update evaluation datasets?
- Can automatically update datasets better discern and differentiate model capabilities?

To this end, we propose two strategies based on cognitive theory, mimicking and extending a given dataset, towards reliable and timely evaluation. i) Our **mimicking** strategy is to leverage LLMs to generate similar ones for each existing sample (marked as seeds), so that we, to the maximum extent, preserve the stylistic and contextual essence of the original data. This is simple and efficient, while it is under-exploration if the data leakage issue will be mitigated. ii) Our **extending** strategy further control the difficulty level of

generated questions according to varying cognitive levels. Here, we borrow the concepts from Bloom's taxonomy, *a hierarchical classification widely used for educational learning objectives into levels of complexity and specificity.* This not only makes our evaluation more systematic, but also leads to a balanced difficulty of dataset, which can better distinguish and analyze the capabilities of models — neither too difficult nor too easy can fairly judge students' learning status.

For evaluation, we systematically investigate the above two strategies regarding effectiveness, stability, and their abilities in dealing with data leakage issue. We generate datasets based on two widely used benchmarks (i.e., MMLU (Hendrycks et al., 2021) and BIG-Bench (bench authors, 2023)), and study six open-source models and four closed-source models. We find that: **1)** Both mimicking and extending strategies show a high level of stability (Section 3.2 & 3.4). **2)** The mimicking strategy is effective to alleviate data contamination. In most cases, compared with the leaked original dataset, there is no significant performance improvement on our updated one (Section 3.3) **3)** While, in the cases that mimicking dataset also shows enhanced performance, our extending strategy can still effectively deal with the data leakage issue (Section 3.4). **4)** We can manipulate the Bloom's concept of sample and the popularity of seeds to control the difficulty of extended dataset. The experimental results also provide fine-grained analysis of LLMs' cognitive levels. Some models demonstrate considerable variations in their performance across different cognitive levels, yet GPT-4 exhibits strong performance across all levels (Section 3.5).

Our main contributions can be summarized:

- We propose two strategies to automate dataset updates to alleviate data leakage issue during evaluation.
- We propose to control the difficulty of dataset with varying cognitive levels, towards fair and fine-grained analysis.
- We have conducted extensive experiments, demonstrating the effectiveness and stability of our strategies.

2 Auto-Dataset Update Framework

In Figure 2, there are two dataset update strategies: mimicking and extending. Both strategies are to generate new samples based on existing ones to mitigate data leakage issue. For each sample, the mimicking setting targets a similar yet unseen sample, while the extending setting aims at further controlling the cognitive levels of the sample generation. Compared with mimicking, extending setting is beyond the scope of the given sample, challenging the model's capabilities in more nuanced and complex scenarios. We introduce these two strategies in Section 2.1 and Section 2.2, respectively. Then, we apply them on widely used benchmarks and manually analyze the quality (Sec 2.4). Finally, we conduct experiments to verify their effectiveness, stability, and ability to deal with the issue (Sec 3).

2.1 Mimicking Strategy

The Mimicking Strategy initiates by iterating through each sample in the selected dataset as a seed. This seed, combined with the corresponding task description, forms the input for prompting LLMs (OpenAI, 2022) to generate new samples. Below is an example (details are in Appendix E):

> Please generate one different but high-quality sample following the task description.
> **[Task description]:** This task evaluate ...
> **[seed sample]:** { "input": "Jamal Murray was perfect from the line", "target_scores": { "plausible": 1, "implausible": 0 } },
> **[new one]:** {"input": "Jamal Murray made 10 three-pointers in a row", "target_scores": { "plausible": 1, "implausible": 0 } },

This strategy is designed to retain the stylistics and knowledge essential in the original ones, thus ensuring the quality of the new sample. After generation, we deploy auto-validation strategies (detail in Appendix A), program check and model consistency check, to filter out answer-incorrect and duplicate samples.

2.2 Extending Strategy

Considering the limited data annotation ability, we adapt Bloom's taxonomy into LLMs evaluation with four cognitive levels. Figure 2 shows them with examples: **Remember and Understand Level** asks the model to list, recall basic concepts, interpret, summarize, and exemplify ideas or concepts; **Apply Level** tests the model to use learned facts and abstractions in new contexts and particular situations; **Analysis Level** requires the model to break down concepts and examine the relationships among them; **Evaluation Level** asks the model to use knowledge and skills to appraise a situation and criticize opinions or statements.

Here to notice, Bloom's taxonomy is a theory

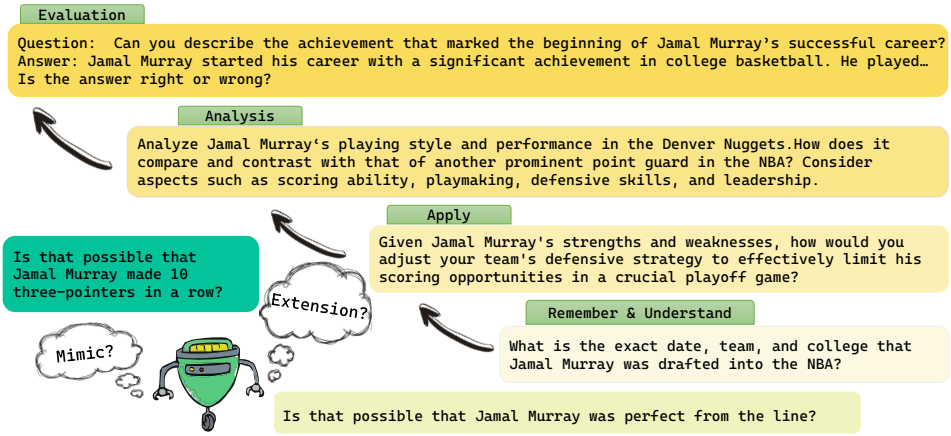

Figure 2: The auto-dataset update framework. For mimicking strategy, we mimic the original test case to get a similar but new sample. For the Extending strategy, we extend the original sample to multi-cognition levels (Remember & Understand , Apply , Analysis , Evaluation) to have a thorough and nuanced assessment.

that categorizes educational objectives into six levels of complexity and specificity: Remember, Understand, Apply, Analyze, Evaluate, and Create. For our purposes, we've combined the Remember and Understand levels for simplicity and excluded the Create level to more clearly distinguish between the different cognitive abilities we're testing.

> Write one sample about the given seed. The question focused on level ...
> **[Seed]:** Jamal Murray
> **[New sample]:** {"question": What is the exact date, team and college that Murray was drafted into the NBA?, "ref_answer": A }

Given that each question in the original dataset is typically aligned with a specific cognitive level, this aspect presents a challenge for extending. To generate multi-level samples effectively, our approach involves abstracting the original question into a core entity, statement, or piece of knowledge. This abstraction can be done either manually or programmatically. Using this abstracted result, which serves as the seed, we guide LLMs to generate new questions focused on different cognitive levels. Above is an example where we extract sports star "Jamal Murray" and prompt LLMs for extending.

It is important to note that our two strategies are efficient and adaptable, making them easily applicable across a wide range of benchmarks.

2.3 Apply to Existing Benchmarks

To conduct the auto-dataset update, we select BIG-Bench (bench authors, 2023) and MMLU (Hendrycks et al., 2021) as our seed dataset. From these benchmarks, we chose ten sub-tasks that are particularly representative of the diverse abilities LLMs are expected to demonstrate. Specifically, from BIG-Bench, we include: **Sports Understanding (Sports):** focuses on the ability to discern between plausible and implausible statements about sports stars. **Periodic Elements (Element):** measures knowledge of chemistry. **CS Algorithms (Algos):** assesses models' understanding and utilization of computer science algorithmic concepts. **Physical Intuition (Phys):** tests the understanding of the physical behaviors **Math Word Problems with Hints (Math):** tests the ability of models to perform mathematical reasoning in the context of arithmetic word problems. From the MMLU, we select five subtasks: **Abstract Algebra (Algebra):** assesses models' understanding of abstract algebra concepts. **International Law (Law):** evaluates model's ability to understand and follow rules and regulations. **Econometrics (Econ):** tests the understanding of econometric principles and implications of econometric phenomena. **College Medicine (Medicine):** evaluates models' knowledge relevant to medicine. **Computer Security (Security):** involves understanding the principles used to protect computer systems and networks.

After selecting datasets, we applied our two strategies to update these datasets and analyze the effectiveness and reliability of the updated samples.

2.4 Updated Dataset Analysis

We utilized ChatGPT (gpt-3.5-turbo) to apply the mimicking strategy for updating selected datasets from BIG-bench and MMLU. It's important to note that following the update process, we conducted a filtering step to remove duplicated and answer-

incorrect samples, as detailed in Appendix A. This filtering process resulted in a dataset size smaller than the original. To achieve a dataset size comparable to the original, we literate the dataset 2-3 times. (considering the time and the cost we limit the amount of the sample from the dataset "Math" to 1000) The statistical details of the updated dataset is shown in Table 1.

Task_{BIG}	#Orig.	#Mimic	$\text{Task}_{\text{MMLU}}$	#Orig.	#Mimic
Sports	1000	951	Algebra	100	93
Element	536	548	Law	121	117
Algos	320	300	Econ	114	101
Phys	81	99	Medicine	172	160
Math	7688	1016	Security	100	100

Table 1: The statistical result of the original (Orig.)and mimicked (Mimic) ten subtasks from BIG-Bench and MMLU. For time and cost consideration, we limited the number of the generated samples on Math.

For the extending strategy, we chose the dataset Sports, Algorithms (Algos), Algebra, and Physics (Phys), because the mimicking strategy performs poorly for data leakage mitigation. For the Sports task, we extract the names of sports stars as seeds, recognizing that this task assesses the model's knowledge of these individuals. For the Algebra task, we extract key algebraic concepts as seeds. For Phys, we use GPT-4 (OpenAI, 2023) to summarize the basic physical laws behind the original sample to use as seeds. Since the Algo task only encompasses a limited range of algorithmic topics, we employed GPT-4 to generate a list of 40 algorithm names. These names were then utilized as seeds for question generation. Based on these seeds, we utilize GPT-4 to generate new samples across multiple cognitive levels using extending prompts (details in Appendix D). We manually maintain an equal distribution of the questions across each cognitive level. The static result is shown in Table 8.

Human Evaluation. To validate the reliability of our two strategies, we conduct a human evaluation involving five senior computational linguistics researchers, who have been trained in advance. For the mimicked samples, evaluators review 120 randomly chosen samples, each including the question's category, the question itself, and the answer. They assess each question and answer paired based on three criteria: Fluency (the grammatical correctness and smoothness of the question), Coherence and Clarity (the logical clarity and explicit articulation of the question), and Accuracy of the Answer (the evaluation guideline is shown in Appendix B). In evaluating the extended samples, evaluators examine 60 randomly selected samples, which include the question, its cognitive level, and the reference answer. The assessment criteria are similar to those for the mimicked samples, with an additional focus on Category Accuracy to ensure that the question's cognitive level was appropriately identified. The human evaluation results, summarized in Table 2, demonstrate high effectiveness and reliability of both the Mimicking and Extend strategies.

Setting	Fluency	Coherence	Answer	Category
Mimicking	94.7 / 95.7	94.4 / 94.0	86.7 / 82.8	-
Extending	98.3 / 100	96.7 / 96.7	92.7 / 82.0	98.3 /100

Table 2: This table presents the overview of Human Evaluation. Results Scores are shown as full score rate (%), with numbers after the slash indicating agreement rates (%) among the five evaluators.

2.5 Evaluation Metric

For mimicked samples, we adhere to the evaluation metrics used in the original tasks (detail in Appendix C.1.5). For extending samples, where questions are free-form, we adopt the "LLM judgment" methodology, following Zheng et al. (2023); Bai et al. (2023). We give the question, reference answer (from updated sample) and the candidate's answer to LLMS to evaluate the answer across three dimensions: 1. **Accuracy:** evaluates the correctness of the answer, 2. **Coherence:** assesses the logical flow, and 3. **Factuality:** assessing the presence of factual errors (prompt are shown in Appendix D). We use the full-mark rate over the three dimensions as the metric and manually evaluate the "LLM judgment" result. The consistency of the model's results with human judgment achieved 90.8% (detailed in Appendix B).

3 Experiment

3.1 Baseline

To empirically validate the reliability of our auto-dataset update framework, we conducted experiments using six open source models: Llama-2-7b-chat, Llama-2-13b-chat (Touvron et al., 2023), Mistral-7B-Instruct-v0.2, Mixtral-8x7B-Instruct-v0.1 (Jiang et al., 2024), Yi-6b-chat, Yi-34b-chat (lingyiwanwu, 2023), and 4 closed-source models: GTP-4, ChatGPT, Claude2 (Anthropic, 2023), Gemini-pro (Google, 2023).

3.2 Evaluation stability of updated datasets

Our proposed strategies are able to update datasets with minimum human effort whenever needed.

Model	Sports$_o$	Sports$_m$	Element$_o$	Element$_m$	Algos$_o$	Algos$_m$	Phys$_o$	Phys$_m$	Math$_o$	Math$_m$
GPT-4	89.9	92.1 ± 1.6	79.9	15.7	25.8	32.3 ± 1.6	85.2	93.2 ± 1.7	85.0	75.1 ± 0.6
ChatGPT	97.3	97.7 ± 0.9	53.9	2.9	23.4	28.7 ± 1.3	79.1	86.3 ± 2.6	40.2	43.1 ± 2.6
Gemini	86.7	90.9 ± 1.8	68.8	15.3	10.3	14.3 ± 1.6	80.2	88.8 ± 2.1	20.8	23.1 ± 1.6
Claude2.1	99.4	98.4 ± 0.9	52.9	8.3	16.6	17.8 ± 2.1	79.1	85.5 ± 2.1	44.5	46.0 ± 0.5
Llama-7b	94.3	94.2 ± 0.3	19.9	3.1	2.0	3.3 ± 0.6	44.4	67.3 ± 1.6	14.6	15.1 ± 2.0
Llama-13b	92.7	95.6 ± 0.4	27.8	4.0	6.1	3.4 ± 0.6	54.3	71.8 ± 2.8	19.6	24.1 ± 0.1
Mistral-7b	88.8	94.0 ± 0.6	27.1	5.3	15.7	20.6± 0.6	53.1	75.4 ± 0.1	12.5	25.8 ± 0.4
Mistral-8x7b	92.0	96.6 ± 0.1	39.7	7.1	10.0	10.1 ±0.0	63.0	82.0 ± 1.7	29.6	35.6 ± 2.2
Yi-6B	50.1	54.1 ± 1.7	21.1	3.1	3.1	2.3 ± 0.3	34.6	42.2 ± 2.4	3.9	3.0 ± 0.0
Yi-34B	62.7	70.5 ± 1.6	25.9	4.2	3.1	4.6 ± 1.3	71.6	84.1 ± 3.7	29.2	31.8 ± 1.1

Table 3: The performance (%) of the involved 10 models (zero-shot) on the original and mimicked BIG-bench. Footnote o indicates the original datasets while m represents the mimicked ones. The experiment of mimicked dataset generation is conducted four times, and the table shows the average performance and the standard deviation.

Model	Algebra$_o$	Algebra$_m$	Law$_o$	Law$_m$	Econ$_o$	Econ$_m$	Medicine$_o$	Medicine$_m$	Security$_o$	Security$_m$
GPT-4	76.0	72.5 ± 1.6	89.7	91.6 ± 0.9	66.7	76.3 ± 2.3	80.7	85.2 ± 1.7	85.0	91.6 ± 0.2
ChatGPT	39.0	37.7 ± 0.1	73.6	80.6 ± 0.0	46.4	55.0 ± 2.4	70.1	73.5 ± 0.2	79.0	81.9 ± 0.2
Gemini	35.0	36.6 ± 0.0	82.6	85.2 ± 0.0	54.4	57.4 ± 0.2	69.3	71.5 ± 1.2	78.0	82.0 ± 0.0
Claude2.1	47.0	52.6 ± 0.2	69.4	70.2 ± 0.1	40.4	54.4 ± 0.2	60.7	62.5 ± 0.1	68.0	75.6 ± 2.4
Llama-7b	14.0	17.2 ± 0.2	57.8	68.0 ± 2.1	28.1	32.3 ± 1.6	41.9	49.8 ± 1.4	58.0	60.6 ± 2.3
Llama-13b	31.0	29.0 ± 0.1	70.0	77.2 ± 0.2	30.0	36.0 ± 1.6	43.6	50.1 ± 1.2	65.0	72.0 ± 1.2
Mistral-7b	34.0	27.9 ± 1.0	70.2	77.1± 0.2	36.9	45.7 ± 0.7	59.5	59.3 ± 0.1	70.0	74.4 ± 2.8
Mistral-8x7b	41.0	41.5 ± 0.4	74.4	77.9 ± 1.0	53.5	62.5 ±0.2	67.6	73.5 ± 1.2	74.0	77.1 ± 2.9
Yi-6B	27.0	26.7 ± 2.3	68.6	73.3 ± 0.2	33.3	43.6 ± 1.6	65.9	61.9 ± 0.3	67.0	72.9 ± 2.1
Yi-34B	33.0	37.6 ± 1.6	81.8	82.6 ± 1.4	52.6	58.9 ± 2.4	67.6	71.9 ± 1.0	77.0	79.7 ± 1.2

Table 4: The average performance (%) of the involved 10 models (zero-shot) on the original and mimicked MMLU. The update is conducted four times, and the table shows the average performance and the standard deviation.

That is, if we apply our strategy to the same dataset several times, we will obtain multiple different datasets. Clearly, a major concern is whether they can produce consistent evaluation results. To answer the above stability question, we choose the mimicking strategy and iterate the update process four times. Similar experiments under the extending strategy will be discussed in Section 3.4. The statistical details can be found in Table 11.

Based on four mimicked datasets, we calculate the average performance and the standard deviation of 10 baseline models under zero-shot in context learning (ICL). As shown in Table 3 and Table 4, it is apparent that: **1)** Compared with the performance of baselines on original datasets, their performance on our curated datasets are similar — the difference of the two scores is 4% on average except Element (we will discuss it later) , 22% max and 0% min. This is mainly because the mimicked dataset has a similar style and difficulty; **2)** Among four different mimicked datasets, the standard deviation is limited, ranging from 0% to 3%. This demonstrates the stability of our mimicking dataset update strategy; **3)** The performance on the "Periodic Elements" in the mimicked dataset is markedly lower than in the original. This discrepancy can be attributed to the inclusion of the periodic table in the task description to assist the model in generating new questions. Unlike the original task, which exhaustively covered all possible one-hop questions like *"What element contains one more proton than hydrogen?"*, the model, with the aid of the periodic table, generated more complex queries such as *"What element contains two more protons than hydrogen?"*, which may need a reasoning step to answer compared to the original sample. This also demonstrates a possibility of introducing external knowledge for updates, which will be future work.

3.3 Can mimicked datasets deal with data leakage issues?

One of our main motivations is to automate dataset updates, so as to mitigate the overestimation caused by data leakage issue — the pre-training procedure includes testing data, highlighted by Zhou et al. (2023). To simulate the data leakage issue, we finetune models using the testing data from original datasets (combined into one testing set) and evaluate them on our updated dataset to see if the performance is affected similarly or not. In specific, we choose three baseline models, Llama-2-7b-chat, Llama-2-13b-chat (Touvron et al., 2023), and Mistral-7b-instruction (Jiang et al., 2023), and follow the training configuration in previous work (Zhou et al., 2023; Touvron et al., 2023; Xu et al., 2023) for fair evaluation (detailed configuration is in the Appendix C). For

Model	Training	LoRa Only	Sports$_o$	Sports$_m$	Element$_o$	Element$_m$	Algos$_o$	Algos$_m$	Phys$_o$	Phys$_m$	Math$_o$	Math$_m$
Llama-7b	None	-	94.3	94.2	19.9	3.1	2.0	2.6	44.4	66.7	14.6	15.1
Llama-7b	+ leakage	✓	99.7	87.4	57.1	0.9	42.2	34.0	74.9	67.6	43.6	18.0
Llama-7b	+ w rationale	✓	92.4	92.7	39.9	1.3	37.2	36.0	67.9	67.7	25.8	19.8
Llama-7b	+ leakage	✗	99.8	85.8	42.9	0.0	32.5	25.3	70.4	65.7	42.0	19.2
Llama-7b	+ w rationale	✗	99.7	88.7	47.8	0.2	40.6	34.0	70.4	68.7	32.4	20.8
Llama-13b	None	-	92.7	96.1	27.8	4.0	6.1	3.4	54.3	74.7	19.6	24.6
Llama-13b	+ leakage	✓	99.8	89.2	65.7	1.1	54.4	42.7	83.9	73.7	43.6	23.1
Llama-13b	+ w rationale	✓	96.5	91.7	49.3	1.8	45.6	43.3	74.0	72.7	32.0	28.6
Llama-13b	+leakage	✗	99.7	88.5	40.7	0.0	36.3	28.0	71.6	66.7	42.3	26.4
Llama-13b	+ w rationale	✗	94.3	88.9	43.6	0.9	37.3	39.4	76.6	73.7	36.4	28.2
Mistral-7b	None	-	88.8	94.0	27.1	5.3	15.7	20.0	53.1	75.4	12.5	25.8
Mistral-7b	+ leakage	✓	99.8	87.7	36.6	1.6	38.1	30.7	66.7	69.7	49.9	24.1
Mistral-7b	+ w rationale	✓	95.0	90.4	54.8	1.6	46.6	40.6	81.0	79.0	34.4	21.5
Mistral-7b	+ w rationale	✗	98.3	88.2	61.0	0.0	45.9	38.0	88.9	63.7	48.0	27.5

Table 5: Finetune performance (%) (zero-shot) of Llama-2-7b-chat, Llama-2-13b-chat and Mistral-7b-instruction on the original and mimicked BIG-bench examples (here we use the dataset from one of the iterations for time and cost consideration), *leakage* denote use test prompt and the test set during training. *w rationale* denote using test set with rational (detail in Sec 3.3). For our fine-tuning process, we specifically employed: full parameter and Lora-only

Model	Training	LoRa Only	Algebra$_o$	Algebra$_m$	Law$_o$	Law$_m$	Econ$_o$	Econ$_m$	Medicine$_o$	Medicine$_m$	Security$_o$	Security$_m$
Llama-7b	None	-	14.0	17.2	57.8	70.0	28.1	32.7	41.9	49.8	58.0	60.0
Llama-7b	+leakage	✓	52.0	31.2	95.9	70.8	65.8	34.6	79.8	50.4	86.0	66.0
Llama-7b	+ w rationale	✓	33.0	30.1	74.4	72.7	38.6	31.7	53.8	50.8	67.0	60.0
Llama-7b	+ leakage	✗	49.0	23.7	93.4	70.9	65.8	33.6	79.2	51.1	84.0	60.0
Llama-7b	+ w rationale	✗	42.0	31.2	81.8	73.5	46.5	36.6	62.4	55.0	76.0	63.0
Llama-13b	None	-	31.0	29.0	70.0	77.8	30.0	36.7	43.6	50.1	65.0	72.0
Llama-13b	+ leakage	✓	51.0	30.1	96.7	79.5	67.5	42.5	83.2	54.9	92.0	76.0
Llama-13b	+ w rationale	✓	34.0	32.6	86.0	80.5	39.5	40.2	58.4	55.5	75.0	74.0
Llama-13b	+ leakage	✗	48.0	23.7	92.6	70.9	60.5	36.6	82.1	50.4	83.0	71.0
Llama-13b	+ w rationale	✗	38.0	35.4	86.7	80.1	50.0	40.5	65.3	55.6	79.0	74.0
Mistral-7b	None	-	34.0	26.8	70.2	77.1	36.9	45.7	59.5	59.3	70.0	74.0
Mistral-7b	+ leakage	✓	63.0	32.6	96.7	75.2	70.2	42.6	90.2	56.3	90.0	75.0
Mistral-7b	+ w rationale	✓	39.0	30.1	90.0	79.3	54.4	46.5	75.7	61.8	78.0	76.0
Mistral-7b	+ w rationale	✗	50.0	26.9	97.5	80.1	68.4	50.8	79.8	62.1	86.0	76.0

Table 6: Finetune performance (%) (zero-shot) of Llama-2-7b-chat, Llama-2-13b-chat and Mistral-7b-instruction on the original and mimicked MMLU examples. Following setting in Table 5.

each model, there are three training settings: **1)** *None*, which denotes the original model; **2)** *leakage*, referring to the finetuned model using the above leakage simulation; **3)** *w rationale*, which included rationales for each leakage sample during finetuning. Thus, these rationale finetuning will also improve the model's reasoning ability over the testing samples. In specific, we follow Wei et al. (2022a) to generate rationales using GPT-4 (OpenAI, 2023) for the correct choice. The detailed prompt is shown in the Appendix D.3.

As shown in Table 5 and Table 6, we find that: **1)** on the original dataset, the performance of all leakage simulations increases dramatically, no matter finetune via LoRA (Hu et al., 2021) or not. This is consistent with previous works (Zhou et al., 2023), showing the serious data leakage issue; **2)** on our mimicked datasets, the performance of all leakage simulations decreases or remains similar, except task Algos, and Algebra (will be discussed with the extending strategy); **3)** in most cases, the performance of *leakage* setting is better than that of *w rationale* on original datasets, while the results are opposite in mimicked datasets. We attribute this to the improvement of the generalization ability through training with rationales, which will depress/mitigate the overfitting performance of *leakage/w rationale*, aligned with Longpre et al. (2023). **4)** Llama-2-13b gets improved from 3.4% to 43.3% on the task Algos after finetuning. We hypothesize this enhancement is attributable to model learning the distribution of labels, considering that the labels range uniformly from 1 to 10. This indicates a limitation of the mimicking strategy: the possibility of data leakage leading to models learning from the answer distribution, which in turn impacts the results on the updated dataset.

3.4 Extending Strategy: Stability & Dealing with Data Leakage

As shown in Section 3.3, the mimicking strategy does not work well on two datasets: Algos and Algebra. The data leakage issue still leads to the overestimation of model performance, which we will further explore using the extending strategy in this section. Following the experiment settings

of the mimicking strategy, we leverage the same baseline models with leakage simulations, as well as four iterations of dataset updates for stability verification (updating process deine in Sec 2.4 and statistical details can be found in Table 12). As shown in Table 7, the standard deviation among four runs is quite low, demonstrating the stability of our extending strategy. This is not surprising as extending strategy can be regarded as an extension of the mimicking strategy with varying cognitive levels for controllable difficulty (Sec 3.5).

For data leakage investigation, we still leverage the baseline models in the mimicking strategy, which are finetuned with the testing samples in original datasets. We can see that from Table 7 all performances of leakage simulations greatly decrease compared with original models due to the overfitting issue. Moreover, the decline of *leakage* is more than that of *w rationale*, because it is alleviated through the improved generalization ability via learning with rationales. But, note that even with rationales, the model still suffers from apparent overfitting, which may improve their performance on the original testing set though.

Model	Training	LoRa only	Algebra	Algos
Llama-7b	None	-	**4.2** ±0.6	**10.2** ±0.5
Llama-7b	+ leakage	✓	1.6 ±0.6	2.2 ±0.6
Llama-7b	+ w rationale	✓	1.2±0.0	5.3 ±0.5
Llama-7b	+ leakage	✗	1.8 ±0.6	1.1±0.0
Llama-7b	+ w rationale	✗	2.1 ±0.5	8.9 ±0.8
Llama-13b	None	-	**8.5** ±0.4	**11.9**±0.8
Llama-13b	+ leakage	✓	6.6 ±0.5	5.1 ±0.1
Llama-13b	+ w rationale	✓	6.4 ±0.9	11.2 ±0.8
Llama-13b	+ leakage	✗	1.5 ±0.4	0.5 ±0.4
Llama-13b	+ w rationale	✗	5.7 ±0.5	7.7 ±1.2
Mistral-7b	None	-	**21.8** ±1.7	**36.8** ±0.0
Mistral-7b	+ leakage	✓	0.9 ±0.5	5.3 ±0.5
Mistral-7b	+ w rationale	✓	4.4 ±0.6	9.1 ±0.5
Mistral-7b	+ w rationale	✗	2.3 ±0.2	4.4 ±0.8

Table 7: Average percentage (%) of full-mark of the fine-tuned model on the extended dataset over the four iterations, and the standard deviation.

3.5 Can we control the sample difficulty using the extending strategy?

Except for data leakage in Section 3.4, there is another issue on task Sports and Phys (Table 5 and Table 6) — the absolute scores are quite high so that models' performance cannot be distinguished well (i.e., 3.82% and 3.46% difference on average for models GPT-4, ChatGPT, Gemini, and Claude). In this section, we explore how to control the difficulty of extended datasets from two aspects: **1)** adjusting the cognitive levels, more abstract levels are expected to produce more difficult samples; and **2)** entity popularity, questions about long-tail entities are much more difficult to answer for LLMs (Mallen et al.). Specifically, we use the extracted sports star name and the summarized common physical law as the seed and update the dataset using extending strategy (Section 2.4). We use models GPT-4, ChatGPT, Gemini, and Claude as the baseline for their indistinguishable performance.

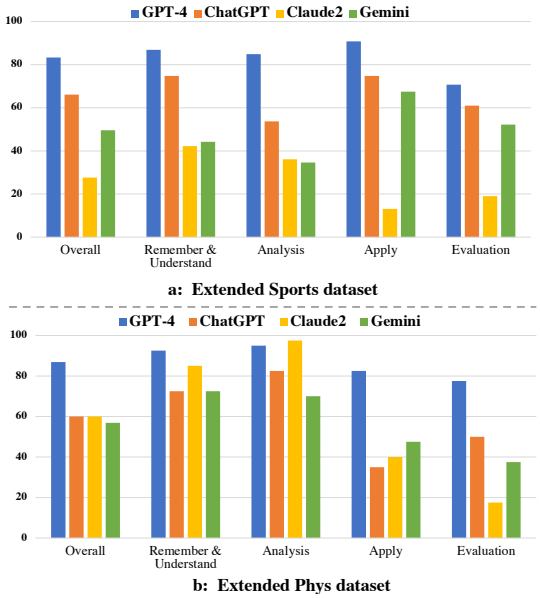

Figure 3: Rull-mark rate (%) of the mode GPT-4, ChatGPT, Claude and Gemini on the extended Sports datasets and Phys on different cognitive levels.

As shown in Figure 3, it is evident that **1)** on the extended Sports and Phys dataset, the four models' performance show larger variance (23.76%, 14.04% respectively), compared with it on original datasets; **2)** overall, samples test "Apply" and "evaluation" cognitive levels are found to be more challenging. This finding indicates that manipulating the cognitive level in the extending strategy is an effective way of controlling the difficulty of the questions. It also highlights the importance of focusing on models' capabilities across different cognitive levels. Some models exhibit significant disparities in performance across various cognitive levels, while GPT-4 shows stronger performance.

In addition to assessing how cognitive levels influence performance, we explored how the popularity of the seeds used in dataset update affects model performance. Following previous work (Mallen et al.), where using Wikipedia page views as a measure of popularity, we gather the total page views for each seed throughout the year 2023 and use it as the popularity. As shown in Figure 4, despite an uneven distribution of seed popularity in the extended dataset, a clear trend emerged: there was

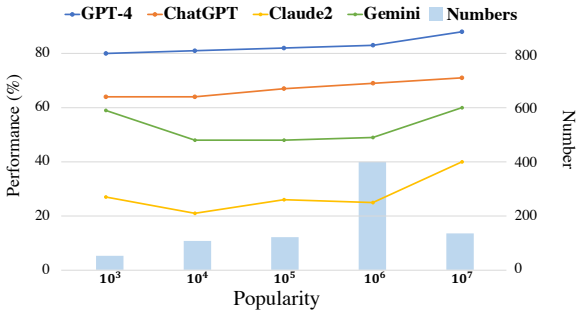

Figure 4: Rull-mark rate (%) of GPT-4, ChatGPT, Claude and Gemini on the extended "sport understanding" datasets on different popularity levels. We depict the number of questions on different popularity levels.

a noticeable increase in model performance when using more popular seeds to get the question. This observation suggests that the popularity of the seed input can also be strategically manipulated to control the difficulty of the generated questions.

3.6 Does extended data samples have data leakage problem?

If the generated samples are from some of the pretrained datasets, it will lead to potential data leakage problems. In this section, we measure the potential data leakage in these extended samples[1]. Following previous work (Li, 2023; Dodge et al., 2021), we calculate the similarity between newly generated data and data from public sources. We involve public data from three sources: web data, public benchmarks, and Instruction Fine-Tuning (IFT) data. For web data, we extracted relevant queries and context through Bing searches API, for public benchmarks we select several popular benchmarks: MMLU, HellaSwag (Zellers et al., 2019), BIG-bench, ARC (Clark et al., 2018), CommonsenseQA (Talmor et al., 2019), Winogrande (Sakaguchi et al., 2021), and for IFT dataset we select LaMini-Instruction (Wu et al., 2023), WizardLM-evol-instruct-V2 (Xu et al., 2023), and P3 (Sanh et al., 2021). Following the methodology in Li (2023), we utilize Meteor (Banerjee and Lavie, 2005) as our similarity metric, and use the same parameter setting in Li (2023). We categorized the generated samples following Dodge et al. (2021): **Clean:** No contamination present. **Input Dirty:** Only the question appears in the public data.**Input-and-Label Dirty:** Both question and answer are found in the public data. As evidenced in Table 8, extending strategy won't have data leakage issue.

[1]Considering the time and cost, we only use the first iteration, which statistics is shown in Table 8)

Dataset	#Total	#Clean	#Input Dirty	#Input-label Dirty
Algebra	80	80	0	0
Algos	320	320	0	0
Phys	80	79	1	0
Sports	824	824	0	0

Table 8: The contamination report on the extended data samples. We compare the generated sample with the web source, the benchmark source, and IFT dataset.

4 Related Work

Benchmark for Model Evaluation: There are a lot open-source benchmarks: MMLU (Hendrycks et al., 2021), tests a wide range of knowledge and reasoning abilities. HellaSwag (Zellers et al., 2019), challenges models with complex commonsense reasoning. BIG-bench (bench authors, 2023), tests models on both traditional NLP tasks and novel problems. ARC (Clark et al., 2018), focusing on science questions that require deep reasoning. KoLA (Yu et al., 2023), also uses the bloom taxonomy to construct dataset. Given that these public benchmarks are open-source and static, they are susceptible to data leakage. **Data Leakage:** Data Leakage could be a crucial problem, study (Zhou et al., 2023) shows that a fine-tuned small model can nearly achieve nearly perfect performance on pen-source benchmarks. To combat this, researchers have developed various detection methods. Study (Dodge et al., 2021) proposed detecting exact matches between test examples and pretraining data. Work (Li, 2023), proposed a contamination report for the open-sources model by using perplexity. The work (Wei et al., 2023) design two indicators using the perplexity to indicate the potential data leakage. However, these detection methods have limitations for they can not be applied to those closed-source models.

5 Conclusion

This paper presents two automated strategies for updating datasets to address data leakage and reduce reliance on labor-intensive manual updates. The mimicking strategy generates new, similar samples based on existing ones, while the extending strategy further controls the difficulty of the generated sample using cogitation levels. Extensive experiments using ten LLMs on updated samples from MMLU and BIG-Bench datasets indicate the effectiveness and stability of our strategies in addressing data leakage. In the future, we are interested in introducing external or domain-specific knowledge for dataset updates.

6 Limitation

While our auto-data update framework involves two strategies to update the dataset to alleviate data leakage, it is important to acknowledge certain limitations: 1. During the creation of the mimic dataset, we filtered out samples where the answers were incorrect. This filtration process may introduce variations in the complexity of the mimicked questions. For instance, models generally exhibit higher performance on mimicked questions on the Phys task. Despite these differences, our human evaluation results, detailed in Section 2.4, confirm the high quality of the mimic dataset. 2. Our strategy of utilizing cognitive levels and entity popularity to adjust question difficulty shows promise. However, there is room for further exploration into more granular methods of setting question difficulty.

References

Anthropic. 2023. Claude 2.1.

Yushi Bai, Jiahao Ying, Yixin Cao, Xin Lv, Yuze He, Xiaozhi Wang, Jifan Yu, Kaisheng Zeng, Yijia Xiao, Haozhe Lyu, et al. 2023. Benchmarking foundation models with language-model-as-an-examiner. *arXiv preprint arXiv:2306.04181*.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

BIG bench authors. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on Machine Learning Research*.

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.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1286–1305.

Google. 2023. Gemini.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav Goldberg. 2023. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. *arXiv preprint arXiv:2305.10160*.

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*.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Changmao Li and Jeffrey Flanigan. 2023. Task contamination: Language models may not be few-shot anymore. *arXiv preprint arXiv:2312.16337*.

Yucheng Li. 2023. An open source data contamination report for llama series models. *arXiv preprint arXiv:2310.17589*.

lingyiwanwu. 2023. Yi.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The flan collection: Designing data and methods for effective instruction tuning.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, Daniel Khashabi, Victor Zhong, Weijia Shi, Wen-tau Yih, Luke Zettlemoyer, et al. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. In *ACL*.

Piotr Mirowski, Kory W Mathewson, Jaylen Pittman, and Richard Evans. 2023. Co-writing screenplays and theatre scripts with language models: Evaluation by industry professionals. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–34.

OpenAI. 2022. Introducing chatgpt.

OpenAI. 2023. Openai: Gpt-4.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. Multitask prompted training enables zero-shot task generalization.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022a. Finetuned language models are zero-shot learners.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022b. Emergent abilities of large language models. *Transactions on Machine Learning Research*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed H Chi, Quoc V Le, Denny Zhou, et al. 2022c. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*.

Tianwen Wei, Liang Zhao, Lichang Zhang, Bo Zhu, Lijie Wang, Haihua Yang, Biye Li, Cheng Cheng, Weiwei Lü, Rui Hu, et al. 2023. Skywork: A more open bilingual foundation model. *arXiv preprint arXiv:2310.19341*.

Minghao Wu, Abdul Waheed, Chiyu Zhang, Muhammad Abdul-Mageed, and Alham Fikri Aji. 2023. Lamini-lm: A diverse herd of distilled models from large-scale instructions. *CoRR*, abs/2304.14402.

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.

Jifan Yu, Xiaozhi Wang, Shangqing Tu, Shulin Cao, Daniel Zhang-Li, Xin Lv, Hao Peng, Zijun Yao, Xiaohan Zhang, Hanming Li, et al. 2023. Kola: Carefully benchmarking world knowledge of large language models. *arXiv preprint arXiv:2306.09296*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*.

Kun Zhou, Yutao Zhu, Zhipeng Chen, Wentong Chen, Wayne Xin Zhao, Xu Chen, Yankai Lin, Ji-Rong Wen, and Jiawei Han. 2023. Don't make your llm an evaluation benchmark cheater. *arXiv preprint arXiv:2311.01964*.

A Data analysis

Tabel 9 shows the detailed information of the selected 10 tasks of BIG-bench and MMLU.

Task	:≡ Label in BIG/ MMLU	Cogitative Level	Validation Method
Sports Understanding	context-free question answer domain specific common sense multiple choice json	Evaluation	- Model-self check
Periodic elements	context-free question answer memorization domain specific free response json	Remember	- Program check
CS Algorithms	context question answer domain specific free response json	Apply	- Program check
Physical Intuition	context-free question answer domain specific multiple choice json	Apply	- Model-self check
Math Word Problems with Hints	context-free question answer logical reasoning multiple choice json	Apply	-Program check
Abstract Algebra	context-free question answer domain specific multiple choice json	Apply Evaluation	-Program check
Internation Law	context-free question answer domain specific multiple choice json	Remember Evaluation	-Program check
Econometrics	context-free question answer domain specific multiple choice json	Remember Apply Evaluation	-Program check
College Medicine	context-free question answer domain specific multiple choice json	Remember Apply	-Program check
Computer Security	context-free question answer domain specific multiple choice json	Remember Apply	-Program check

Table 9: This table elaborates on the detailed task information, question format, and corresponding cognitive levels of the selected tasks from BIG-bench and MMLU (for MMLU we manually labeled each task). Additionally, it outlines the validation method employed for the mimicked tasks. For the "Model-self" check, we deploy the model itself to identify and filter out instances with incorrect answers.

B Human Evaluation

B.1 Mimiced Benchmark Evaluation Guideline

To validate the quality of the mimicked data sample, we randomly selected 120 data samples from the mimicked BIG-bench and MMLU. and Then we conducted a human evaluation involving five senior computing language researchers. Evaluators are provided with samples that have: 1) the question category, 2) the question, and 3) the answer. The evaluation guideline is provided below:

This document is a guide for evaluating test questions generated by Mimic, including the category of the question, the question itself, and the answer. You are tasked with evaluating, Please score based on the following criteria:

For the question, using the following metric for scoring:

Fluency: Assess whether the language of the question is fluent, without grammatical or spelling errors. The scoring range is 1-3:
- 1 point: The text contains multiple grammatical and/or spelling errors, significantly impacting the readability and understanding.
- 2 points: The text contains a few grammatical or spelling errors, slightly affecting readability, but the overall meaning of the text is understandable.
- 3 points: The text is grammatically and orthographically correct, expressing fluently and naturally, easy to understand.

- **Coherence and Clarity**: Assess whether the question is logically clear and articulated explicitly. The scoring range is 1-3:
  - 1 point: The question or answer lacks logical structure, is expressed in a disorganized manner, making it difficult for readers to understand.
  - 2 points: The question or answer has a basic logical structure, with a relatively clear theme or argument, but the expression may not be direct enough or some parts may be slightly vague, affecting overall clarity.
  - 3 points: The question or answer has a clear structure, is logically coherent, expressed directly and clearly, easy to understand, and effectively conveys the theme or argument.

- **Highly misleading**: The context strongly influences you to choose the answer.

For the answer, evaluate the Accuracy:

Accuracy: Assess the options and answer. The scoring range is 1-3
- 1 point: The information in the answer does not match the question's requirements, contains serious errors or misleading information, does not answer the question at all, or the options do not have a correct answer.
- 2 points: The answer attempts to address the question but contains partial errors or misses important information.
- 3 points: The answer is completely accurate, appropriately addressing the question.

B.2 Extend Benchmark Evaluation Guideline

To validate the quality of the extended data sample and the evaluation result, we randomly selected 60 data samples from the extended Sports, Algebra, Algos, and Phys tasks. Then we conduct a human evaluation involving five senior computing language researchers. Evaluators are provided include: 1) question, 2) cognitive level, 3) reference answer, 4) candidate response, and 5) evaluation result. The evaluation guideline is provided below:

- This document is a guide for evaluating test questions generated by Extend method, along

with the evaluation result. Each line includes the question "level", the corresponding question, the reference answer, the candidate response, and the evaluation result. Evaluation should be conducted according to the following requirements:

For the question, using the following metric for scoring:

1. Fluency: Assess whether the language of the question is fluent, without grammatical or spelling errors. The scoring range is 1-3:

- 1 point: The text contains multiple grammatical and/or spelling errors, significantly impacting the readability and understanding.

- 2 points: The text contains a few grammatical or spelling errors, slightly affecting readability, but the overall meaning of the text is understandable.

- 3 points: The text is grammatically and orthographically correct, expressing fluently and naturally, easy to understand.

- **2. Coherence and Clarity:** Assess whether the question is logically clear and articulated explicitly. The scoring range is 1-3:

  - 1 point: The question or answer lacks logical structure, is expressed in a disorganized manner, making it difficult for readers to understand.

  - 2 points: The question or answer has a basic logical structure, with a relatively clear theme or argument, but the expression may not be direct enough or some parts may be slightly vague, affecting overall clarity.

  - 3 points: The question or answer has a clear structure, is logically coherent, expressed directly and clearly, easy to understand, and effectively conveys the theme or argument.

- **3. Category Accuracy:** Based on cognitive theory, questions are categorized as follows:

  - Remember and Understand Level asks the model to list, recall basic concepts, interpret, summarize, and exemplify ideas or concepts.

  - Apply Level tests the model to use learned facts and abstractions in new contexts and particular situations.

  - Analysis Level requires the model to break down concepts and examine the relationships among them.

  - Evaluation Level asks the model to use knowledge and skills to appraise a situation and criticize opinions or statements.

  Score the category accuracy considering if the question category matches the content, ranging from 1-3:

  - 1 point: The category does not match the question content at all. The selected category has no relevance to the question's theme or the type of information required.

  - 2 points: The category partially matches the question content. The selected category is somewhat related to the question but is not the best or most accurate category.

  - 3 points: The category perfectly matches the question content. The selected category precisely reflects the content of the question.

For the provided candidate reponse, you are tasked with evaluating the candicate answers

according to three distinct metrics: Accuracy, Coherence, and Factuality. Each of these metrics is critical for assessing the quality of the answers. Below is a detailed scoring guide for each:

1. Accuracy: - 1-3 points: Score from 1 to 3 based on whether the answer is completely inaccurate, partially accurate but contains some errors or misses important information, or is completely accurate, appropriately, and comprehensively answering the question without any errors or omissions.

2. Coherence: - 1-3 points: Score based on whether the answer lacks logical structure and is expressed in a disorganized manner, has basic logical structure and clarity with some unclear or vague parts, or has a clear structure, is logically coherent, expressed clearly, and effectively conveys the theme or argument.

3. Factuality: - 1-3 points: Score based on whether the answer contains multiple factual errors, generally conforms to facts but contains minor errors or inaccuracies, or is entirely based on facts with all provided information being accurate.

Compare your scores for accuracy, coherence, and factuality with the evaluation results for **Consistency**, ranging from 1-3:

- 1 point: The model's evaluation results significantly diverge from the actual quality of the question and answer, showing a high degree of mismatch.
- 2 points: The model's evaluation results reflect the quality of the question and answer to some extent but have some inconsistencies.
- 3 points: The model's evaluation results are highly consistent with the quality of the question and answer, accurately reflecting its performance.

B.3 Huaman Evaluation result

Setting	Fluency	Coherence	Answer Accuracy	Category Accuracy	Evaluation Consistency
Mimic	94.7 / 95.7	94.4 / 94.0	86.7 / 82.8	-	-
Extend	98.3 / 100	96.7 / 96.7	92.7 / 82.0	98.3 /100	90.8 / 86.7

Table 10: This table presents the overview of Human Evaluation. Results Scores are shown as full score rate (%), with numbers after the slash indicating agreement rates (%) among the five evaluators.

Shown in Table 10, human evaluation result indicates over 85% full score rate for each metric. This demonstrates our framework's capability to support reliable automatic question generation.

C Experiment detail

C.1 Model Setting

C.1.1 Question Generation

For generating questions, we configure both ChatGPT and GPT-4 with a temperature setting of 0 and a maximum token length of 512 to ensure precise and deterministic output.

C.1.2 Answer Generation

Answer generation across the 10 models involved is conducted in a zero-shot setting, with all models set to a temperature of 0 and a maximum token length of 1024. The specific prompt used for generating answers is detailed in Appendix D.

C.1.3 Lora Fine-tuning

For Lora fine-tuning, we explore a range of configurations, adjusting the rank (from 32 to 1024 for the 7b and 13b models), learning rate (from 10^{-4} to 10^{-6}), number of epochs (from 1 to 3), and batch size (from 2 to 32). The optimal configuration for each model is determined through a grid search strategy.

C.1.4 Full Parameter Fine-tuning

Similar to Lora fine-tuning, full parameter fine-tuning involves adjusting the learning rate (from 10^{-4} to 10^{-6}), number of epochs (from 1 to 3), and batch size (from 2 to 32), with the best performance for each model selected via a grid search strategy.

C.1.5 Detailed Metric for mimicking

For the mimic setting, we adhere to the evaluation metrics used in the original tasks considering the similar sample format. Specifically, for multi-choice questions, we utilize the Exact Match (EM) metric to gauge accuracy, for free response questions we Exact Match (EM) metric to gauge accuracy.

C.1.6 Detailed Metric for extending

In the extending setting, where questions are free-form, we adopt the "LLM judgment" methodology (Zheng et al., 2023; Bai et al., 2023). We evaluate model performance across three dimensions: 1. **Accuracy:** evaluates the correctness of the answer, 2. **Coherence:** assesses the logical flow and clarity of the response, and 3. **Factuality:** assessing the presence of factual errors in the response (detailed prompt are shown in Appendix D). We use the full-mark rate over the three dimensions as a metric to label the responses.

C.1.7 Model evaluation

In the extending setting, where questions are free-form, we adopt the "LLM judgment" methodology (Zheng et al., 2023; Bai et al., 2023). We deploy GPt-4 as the evaluator, setting temperature as 0, and max_length as 1024.

C.2 More Experiment Result

Task_{BIG}	#Orig.	#Mimic	$\#\text{Mimic}_1$	$\#\text{Mimic}_2$	$\#\text{Mimic}_3$	$\text{Task}_{\text{MMLU}}$	#Orig.	$\#\text{Mimic}_1$	$\#\text{Mimic}_2$	$\#\text{Mimic}_3$	$\#\text{Mimic}_4$
Sports	1000	951	966	958	950	Algebra	100	93	90	95	96
Element	536	548	-	-	-	Law	121	117	119	118	116
Algos	320	300	300	300	300	Econ	114	101	94	99	103
Phys	81	99	99	98	95	Medicine	172	160	168	165	163
Math	7688	1016	1016	978	985	Security	100	100	93	95	98

Table 11: The whole statistical result of the original (Orig.) and four mimicked (Mimic) datasets from BIG-Bench and MMLU. We use mimic strategy to update four times to show the stability of our strategy

Task_{BIG}	$\#\text{Extending}_1$	$\#\text{Extending}_2$	$\#\text{Extending}_3$	$\#\text{Extending}_4$
Algebra	80	80	80	80
Algos	320	320	320	320
Phys	80	-	-	-
Sports	824	-	-	

Table 12: The whole statistical result of the extending datasets from BIG-Bench and MMLU. We use extending strategy to update four times to show the stability of our strategy

D Prompt

D.1 Mimic question

- You are a question-writer expert. Please generate one **different** but high-quality sample following the task description.

  ###Task description###:
  [Your Task description]

  Here is an example, help me generate one **different** but **similar** one, and guarantee the answer is correct.
  [Example from the original datasets]

D.2 Extend question

“Remember and Understand” level
I want you to act as a question writer expert, specializing in the "Remember and Understand" level of cognitive assessment. Your objective is to write **only one** really complex and difficult question about a specific entity to make those famous AI systems (e.g., ChaGPT and GPT4) a bit harder to handle.

[Generate Criterion]
1. The question should be focused on the remember and understand level. This means the question should prompt for recall of facts, terms, and basic concepts, interpret, summarize, and exemplify ideas or concepts. NOT delve into deeper levels like Applying Analyzing or Evaluation.
2. Ensure that you can confidently answer the questions you are proposing, if you can not answer it correctly or have no related knowledge about the entity please return "None".
3. DO NOT add other words other than the question itself.

[Example question]
The example question

The seed enetity
Help me generate the answer also

“Apply” level
I want you to act as a question writer expert, specializing in the "Applying" level of cognitive assessment. Your objective is to write **only one** really complex and difficult question about the given statement to make those famous AI systems (e.g., ChaGPT and GPT4) a bit harder to handle.

[Generate Criterion]
1. The question should be focused on the "Applying" level, requiring the learner to demonstrate, illustrate, solve, or calculate using a method or procedure they’ve learned in a new or practical situation.
2. Ensure that you can confidently answer the questions you are proposing, if you can not answer it correctly or have no related knowledge about the entity please return "None".
3. DO NOT add other words other than the question itself.

The seed statement

Help me generate the answer also

"Analysis" level
I want you to act as a question writer expert, specializing in the "Analysing" level of cognitive assessment. Your objective is to write **only one** really complex and difficult question about a given statement to make those famous AI systems (e.g., ChaGPT and GPT4) a bit harder to handle.

[Generate Criterion]
1. The question should be focused on the "Analysing" level, requiring the learner to break information into parts to explore understandings and relationships. It's about asking learners to look into the components, analysis of relationships, and comparison with other entities or concepts.
2. Ensure that you can confidently answer the questions you are proposing, if you can not answer it correctly or have no related knowledge about the entity please return "None".
3. DO NOT add other words other than the question itself.

The seed statement
Help me generate the answer also

"Evaluation" level
I want you to act as a question writer expert, specializing in the "Evaluation" level of cognitive assessment. Your objective is to write **only one** really complex and difficult question about a specific entity with **an answer** that is difficult to discern, especially for AI systems.

[Generate Criterion]
1. The **answer** provided should be **exceptionally misleading**, making it difficult for even AI systems to differentiate if the answer is correct.
2. Ensure that you can confidently answer the questions you are proposing, if you can not answer it correctly or have no related knowledge about the entity please return "None".
3. DO NOT add other words other than the question itself.

The seed entity
Help me generate the answer also

D.3 Generate rational for model finetune

You are a good assistant and always follow my word.

I will get your one ###Question### and one ###Answer###. The answer is always correct. help me generate an explanation for it return as

Explanation###

###Question###
Input Question

###Answer###
Input Answer

D.4 Question answer

For mimic setting we use the prompt from the original BIG-bench and MMLU, for extend setting the prompt is:

You are a good assistant, please help me answer the question.

D.5 LLM Evaluation

You are a critical assessment expert, and you will be given a set of question-answer pairs. Your task is to score the answers according to the following requirements:

[Evaluation Steps]
a. You should score the answer based on the reference answer.
b. You should rate the answer on 3 metrics, and assign a score between 1 and 3, with 3 being the highest.

[Evaluation Criterion]
1. For accuracy, you will score whether the answer correctly and comprehensively answers the question.
2. For coherence, you will assess the structure and logic of the answer, and whether the answer is understandable by non-professionals.
3. For factuality, you will only evaluate whether the answer contains factual errors.
4. Return in formate: Accuracy: 1

[Reference Answer] *Reference Answer*

E Case Study

E.1 Question Generation

E.1.1 Mimic

Input case:
"input": "a man has some hens and cows . if the number of heads be 50 and the number of feet equals 144 , then the number of hens will be", "hint": "divide(subtract(multiply(50, 4), 144), 2)", "target_scores": "22": 0, "24": 0, "26": 0, "20": 0, "28": 1 ,

Output case:
"input": "In a farm, there are ducks and sheep. If there are 30 heads in total and the number of legs is 100, how many ducks are there?", "hint": "divide(subtract(multiply(30, 4), 100), 2)", "target_scores": "10": 1, "15": 0, "20": 0, "25": 0, "30": 0 ,

Input case:
"question": "Which of the following is a remote Trojan?", "A": "Troya", "B": "DaCryptic", "C": "BankerA", "D": "Game-Troj", "answer": "A" ,

Output case:
"question": "Which of the following is a form of ransomware?", "A": "Jigsaw", "B": "Powload", "C": "Loki", "D": "Kovter", "answer": "A" ,

Input case:
"question": "Which of the following is a remote Trojan?", "A": "Troya", "B": "DaCryptic", "C": "BankerA", "D": "Game-Troj", "answer": "A" ,

Output case:
"question": "Which of the following is a form of ransomware?", "A": "Jigsaw", "B": "Powload", "C": "Loki", "D": "Kovter", "answer": "A" ,

Input case:
"input": "Jamal Murray was perfect from the line", "target_scores": "plausible": 1, "implausible": 0 ,

Output case:
"input": "Jamal Murray made 10 three-pointers in a row", "target_scores": "plausible": 1, "implausible": 0 ,

Input case:
"input": "What element contains one more proton than Hydrogen?", "target_scores": "helium" ,

Output case:
"input": "What element contains two more protons than hydrogen?", "target_scores": "lithium" ,

E.1.2 Extend

Remember and Understanding What is the formal definition of a ring in abstract algebra, including the properties and operations it must possess?
Apply Given a ring R with unity and an element a in R such that a^2 = a, how would you construct a subring of R generated by a?
Analysis Analyze the structure of a ring in abstract algebra. How does it differ from a group and a field in terms of its operations and elements? What are the key components that define a ring and how do these components interact with each other?
Evaluation Question: In the field of abstract algebra, a ring is a set equipped with two binary operations that generalize the arithmetic operations of addition and multiplication. With this in mind, is it correct to say that in any ring, the distributive law always applies in both directions, meaning that for all a, b, and c in the ring, a * (b + c) = a * b + a * c and (b + c) * a = b * a + c * a? Answer: Yes, it is correct. In any ring, the distributive law applies in both directions. This is a fundamental property of rings and is part of the definition of a ring in abstract algebra. The operations of addition and multiplication in a ring are always distributive over each other in both directions.

E.2 LLM Evaluation

Question What was the exact date, college, and 40-yard dash time of DK Metcalf when he was drafted into the NFL?
Reference Answer DK Metcalf was drafted into the NFL on April 26, 2019, from the University of Mississippi (Ole Miss), and his 40-yard dash time was 4.33 seconds.
Reponse DK Metcalf was drafted into the NFL on April 26, 2019. He played college football at the University of Mississippi (Ole Miss). During the NFL Combine, Metcalf ran the 40-yard dash in 4.33 seconds.
Evaluation result Accuracy: 3 - The answer correctly and comprehensively answers the question. Coherence: 3 - The answer is structured logically and is understandable. Factuality: 3 - The answer does not contain any factual errors.
Return format: Accuracy: 3, Coherence: 3, Factuality: 3

Question Analyse the components of a linear regression model, including the dependent variable, independent variables, error term, and coefficients. How do these components interact to predict the dependent variable, and how does this differ from the components and interactions in a logistic regression model?

Reference Answer In a linear regression model, the dependent variable is the outcome we are trying to predict or explain. The independent variables are the predictors or factors that we believe have an impact on the dependent variable. The coefficients represent the change in the dependent variable for a one-unit change in the predictor variable, assuming all other variables are held constant. The error term captures the variability in the dependent variable that is not explained by the predictors.

The components interact in the following way: The predicted value of the dependent variable is equal to the sum of the product of each independent variable and its corresponding coefficient, plus the error term.

In contrast, a logistic regression model also has dependent and independent variables, coefficients, and an error term. However, the dependent variable in logistic regression is binary (0 or 1). The coefficients represent the change in the log-odds of the dependent variable for a one-unit change in the predictor variable. The predicted value is the probability that the dependent variable equals 1, given the values of the independent variables. This is calculated using the logistic function of the sum of the product of each independent variable and its corresponding coefficient, plus the error term.

So, while both models involve a combination of variables and coefficients, the key difference lies in the nature of the dependent variable (continuous vs. binary) and the function used to combine the variables and coefficients (identity vs. logistic).

Reponse A linear regression model consists of a dependent variable, independent variables, error term, and coefficients. The dependent variable is the variable that is being predicted, and the independent variables are the variables that are used to predict the dependent variable. The error term is the difference between the predicted value and the actual value. The coefficients are the weights assigned to each independent variable.

Evaluation result Accuracy: 2 Coherence: 3 Factuality: 3

The answer correctly identifies the components of a linear regression model: dependent variable, independent variables, error term, and coefficients. However, it does not explain how these components interact to predict the dependent variable, which is crucial for a comprehensive answer. Additionally, the answer does not address the second part of the question regarding how this differs from a logistic regression model. The explanation is coherent and easily understandable, and there are no factual errors in the information provided.

E.3 Cases for fine-tuned model vs. original model on the extended dataset

Question What are the two binary operations that define a mathematical structure as a Ring, and what are the four properties that these operations must satisfy?
Level: Remember
Reponse from fine-tuned Mistral (Lora) Answer: addition and multiplication The two binary operations that defined a mathematical structure as a ring are addition and multiplication. For a ring to exist, these operations must satisfy four properties:
1. Closure: For any two elements a and b in the ring, the sum a + b and the product a * b must also be in the ring.
2. Associativity: The addition and multiplication must be associative, meaning (a + b) + c = a + (b + c) and (a * b) * c = a * (b * c) for any elements a, b, and c in the ring.
3. Distributivity: The addition and multiplication must be distributed over each other, meaning a * (b + c) = a * b + a * c and (a + b) * c = a * c + b * c for any elements a, b, and c in the ring.
4. identity: There must be two elements, 0 and 1, in the ring such that for any element a in the ring, a + 0 = a and a * 1 = a.
If a ring satisfies these properties, it is a well- defined structure that can be used in certain types of equations and equations.
Evaluation result Accuracy: 3 Coherence: 3 Factuality: 2
The answer correctly identifies addition and multiplication as the two binary operations that define a ring. It also accurately lists three of the four properties that these operations must satisfy: closure, associativity, and distributivity. However, the fourth property is not completely correct. While the answer correctly states the existence of an additive identity (0), it incorrectly states the existence of a multiplicative identity (1) as a requirement for all rings. In fact, a ring only requires an additive identity; a multiplicative identity is required for a ring to be a commutative ring with unity, but not all rings have a multiplicative identity. Therefore, the answer contains a factual error regarding the properties of a ring. The answer is coherent and well-structured, making it understandable to non-professionals.
Reponse from Mistral A mathematical structure is called a ring if it has two binary operations defined on it, namely addition (+) and multiplication (×), subject to the following conditions:
1. The addition operation must be associative, meaning that for all a, b, and c in the ring, we have: a + (b + c) = (a + b) + c
2. The addition operation must have an identity element, denoted by 0, such that for all a in the ring, we have: a + 0 = a
3. Every element in the ring must have an additive inverse, denoted by -a, such that for all a in the ring, we have: a + (-a) = 0
4. The multiplication operation must be associative, meaning that for all a, b, and c in the ring, we have: a × (b × c) = (a × b) × c 5. The multiplication operation must distribute over addition, meaning that for all a, b, and c in the ring, we have: a × (b + c) = (a × b) + (a × c) and (b + c) × a = (b × a) + (c × a) These properties ensure that the ring has well-defined addition and multiplication operations that behave in a predictable and consistent way.
Evaluation result Accuracy: 3 The answer correctly identifies the two binary operations that define a ring (addition and multiplication) and accurately lists the properties these operations must satisfy: associativity of addition, existence of an additive identity, existence of additive inverses, associativity of multiplication, and the distributive property of multiplication over addition. Coherence: 3 The answer is well-structured and logically presents the information in a clear and understandable manner. It sequentially lists the properties and explains them in a way that is accessible to non-professionals. Factuality: 3 The answer contains no factual errors. It correctly states the properties that define a ring in mathematics. Final Evaluation: Accuracy: 3 Coherence: 3Factuality: 3

Question If you are given two prime numbers, 13 and 17, how would you generate a public and private key pair using the RSA algorithm?
Level: Apply
Reponse from fine-tuned Mistral (Full Parameter) Answer: To generate a public and private key pair using the RSA algorithm, we need to follow the steps outlined in the algorithm.
1. **Choose a modulus**: The modulus is a number that is divisible by both the public and private keys. In this case, we are given the prime numbers 13 and 17. The largest number that is divisible by both 13 and 17 is 17. Therefore, we can use 17 as the modulus.
2. **Generate the public key**: The public key is the key that is made available to anyone who wants to verify the authenticity of the sender's messages. It is generated by multiplying the modulus by the sender's public exponent (which is 1 in this case). So, the public key is 17 * 1 = 17.
3. **Generate the private key**: The private key is the key that is kept secret by the sender and is used to encrypt and decrypt messages. It is generated by multiplying the modulus by the sender's private exponent (which is also 1 in this case). So, the private key is 17 * 1 = 17.
Therefore, the public and private key pair that can be generated using the RSA algorithm with the prime numbers 13 and 17 is 17 (public key) and 17 (private key). **Evaluation result** Accuracy: 1 Coherence: 2 Factuality: 1
The answer provided contains several inaccuracies and misunderstandings of the RSA algorithm:
1. The modulus (n) is not the largest number divisible by both primes, but rather the product of the two primes. In this case, n should be 13 * 17. 2. The public key (e) is not simply the modulus multiplied by 1. It is a value chosen such that 1 < e < ϕ(n), where ϕ(n) is the totient of the modulus (n), and e is coprime to ϕ(n).3. The private key (d) is not the modulus multiplied by 1. It is calculated such that (d * e) % ϕ(n) = 1, meaning d is the modular multiplicative inverse of e modulo ϕ(n). The answer is coherent in that it presents the steps in a structured manner, but the logic is flawed due to the misunderstandings of the RSA algorithm. The answer contains factual errors regarding the RSA algorithm and the process of generating keys.
Reponse from Mistral
To generate an RSA key pair from two given prime numbers (p = 13 and q = 17), follow these steps:
1. Calculate n = p * q = 13 * 17 = 221.
2. Find ϕ(n) = (p-1) * (q-1) = 12 * 16 = 192.
3. Choose a public encryption exponent e, which must be relatively prime to ϕ(n). A common choice is e = 17 (since 17 is also a prime number and 17 is relatively prime to 192).
4. Calculate the private decryption exponent d, which satisfies the equation d * e % ϕ(n) = 1. Using the extended Euclidean algorithm, we can find d = 113.
5. Your public key is (e, n), and your private key is (d, n). Now you have a public key (e, 221) and a private key (d, 221). You can use the RSA encryption algorithm to encrypt messages using the public key and decrypt them using the private key.
Evaluation result Accuracy: 3 The answer correctly identifies the two binary operations that define a ring (addition and multiplication) and accurately lists the properties these operations must satisfy: associativity of addition, existence of an additive identity, existence of additive inverses, associativity of multiplication, and the distributive property of multiplication over addition. Coherence: 3 The answer is well-structured and logically presents the information in a clear and understandable manner. It sequentially lists the properties and explains them in a way that is accessible to non-professionals. Factuality: 3 The answer contains no factual errors. It correctly states the properties that define a ring in mathematics. Final Evaluation: Accuracy: 3 Coherence: 3 Factuality: 3