



FROG: Evaluating Fuzzy Reasoning of Generalized Quantifiers in Large Language Models

Yiyuan Li^{*}

Shichao Sun^{*}

Pengfei Liu^{*}

Shanghai Jiao Tong University^{*}

UNC-Chapel Hill^{*}

The Hong Kong Polytechnic University^{*}

yiyuanli@cs.unc.edu, ccssun@comp.polyu.edu.hk, pengfei@sjtu.edu.cn

Abstract

Fuzzy reasoning is vital due to the frequent use of imprecise information in daily contexts. However, the ability of current large language models (LLMs) to handle such reasoning remains largely uncharted. In this paper, we introduce a new benchmark, FROG, for fuzzy reasoning, featuring real-world mathematical word problems that incorporate generalized quantifiers. Our experimental findings reveal that fuzzy reasoning continues to pose significant challenges for LLMs. Moreover, we find that existing methods designed to enhance reasoning do not consistently improve performance in tasks involving fuzzy logic. Additionally, our results show an inverse scaling effect in the performance of LLMs on FROG. Interestingly, we also demonstrate that strong mathematical reasoning skills are not necessarily indicative of success on our benchmark¹.

1 Introduction

The capability to reason effectively is a critical dimension in evaluating the advancements of large language models (LLMs) (Brown et al., 2020; Huang and Chang, 2023; Bowen et al., 2024; Wang et al., 2024b; Zhu et al., 2024). Commonly, the progress of advancing reasoning abilities is mainly evaluated on mathematical reasoning benchmarks (Xia et al., 2024; Huang et al., 2024a) like GSM8K (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021), which demands **precise** answers derived from clear, rule-based questions. However, much of human knowledge and daily decision-making processes are not based on precision but rather involve handling **fuzzy**, imprecise information (Novák, 2015).

Fuzzy reasoning, which deals with uncertainties and perceptual data, differs significantly from the deterministic processes evaluated in existing

standard benchmarks. It often relies on natural language constructs that do not capture information with precise granularity (Novák, 2015). For instance, generalized quantifiers (GQs), such as “few” or “most”, are frequently used in natural language to introduce vagueness (Mostowski, 1957; Ramotowska et al., 2024). An illustrative fuzzy problem employing GQ might be: “*There have been successive increases of 20% and then most in the price of gas from the previous month. By what percentage should a driver reduce gas consumption so that the expenditure does not change?*” Here, the term “*most*” introduces ambiguity concerning the extent of the price increase and necessitates an estimation of its semantics to solve the problem accurately.

However, such GQ-based fuzzy reasoning problems are under-explored. Faghihi et al. (2020) introduces probabilistic fuzzy logic (Yager and Zadeh, 1992) to enhance reasoning abilities. Existing works like Saki and Faghihi (2022) investigate employing fuzzy probabilistic theory in association-based problems, like the relation between fuzzy features and labels (e.g. *a lot of* smoking and cancer). The mapping functions to encode fuzzy features are either simulated (Faghihi et al., 2020) or collected from limited data with heuristics (Wang et al., 2019). Those mapping functions fail to evaluate complex, real-world reasoning scenarios and realistically capture how LLMs utilize fuzzy reasoning in practical contexts.

In this paper, we aim to explore reasoning challenges associated with fuzzy events (Zadeh, 1968), which are mathematically ambiguous and articulated through fuzzy expressions of GQs, such as “most in the price of gas”. Specifically, we focus on mathematical reasoning problems that incorporate GQs. To this end, we have developed FROG, a benchmark for **F**uzzy **R**easoning benchmark of **G**eneralized quantifiers, which transforms real-world mathematical problems from GSM8K (Cobbe et al., 2021) and MathQA (Amini

^{*}Corresponding author

¹Resource: <https://github.com/Nativeatom/FROG>

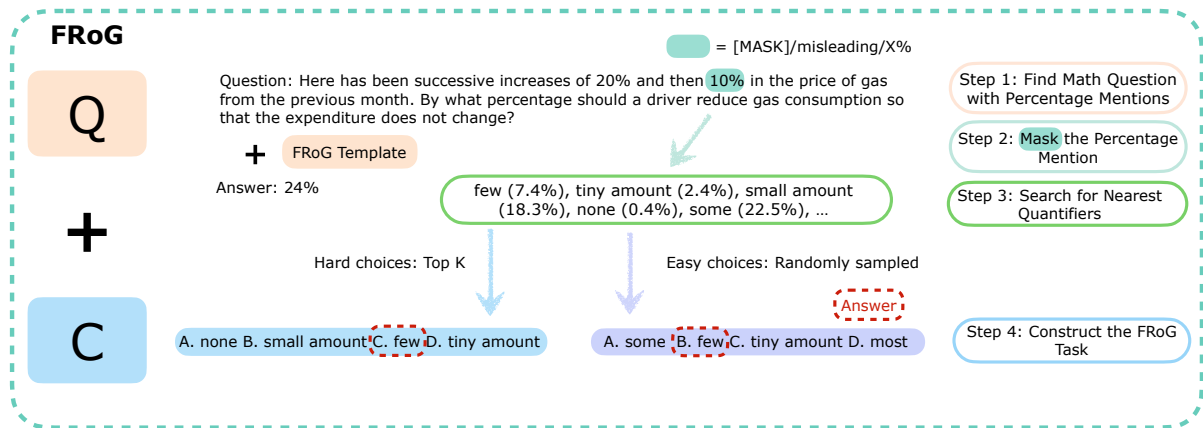


Figure 1: Workflow of FROG construction.

et al., 2019) datasets into multiple-choice questions. These questions replace precise numerical data with GQs, requiring models to apply both precise reasoning like arithmetic computations and fuzzy reasoning to estimate the scope of GQs. Our evaluation of several LLMs reveals that fuzzy reasoning remains a significant challenge in FROG, with an inverse scaling effect observed in over half of the model families tested. Furthermore, common methods designed to enhance reasoning capabilities like math-specialized tuning, code-specialized tuning, and general alignment, do not consistently improve performance in FROG. Interestingly, we find that strong mathematical reasoning skills are not necessarily indicative of success on FROG. We also outline the primary strategies employed by LLMs to perform fuzzy reasoning.

2 Related Work

Reasoning abilities, involving drawing conclusions from existing knowledge, are a cornerstone of human intelligence and are crucial for intricate tasks like decision-making and solving math word problems (Yu et al., 2023). Recently, mathematical problem-solving has become a key dimension in assessing the reasoning capabilities of LLMs (Xia et al., 2024; Huang et al., 2024a). Various approaches have been developed to enhance the math reasoning ability of LLMs, like prompt-based designs (Chia et al., 2023; Zheng et al., 2023; Chen et al., 2023a). On the other hand, Taylor et al. (2022); Lewkowycz et al. (2022); Paster et al. (2024); Azerbayev et al. (2024) propose data construction for the pertaining stage. And further supervised fine-tuning, instruction tuning or alignment methods like Direct Preference Optimization

(DPO, Rafailov et al. (2023)) are used to enhance the reasoning abilities of LLMs (Yu et al., 2024; Luo et al., 2023; An et al., 2023; Huang et al., 2024b; Li et al., 2024; DeepSeek-AI et al., 2024). Motivated by the fact that external tools are widely used in NLP tasks, tool integration is introduced to enhance math reasoning (Mishra et al., 2022; Gao et al., 2023; Gou et al., 2024a,b; Yue et al., 2024; Zhou et al., 2024; Zhang et al., 2024a). Chen et al. (2023b) employs programming methods to enhance the reasoning ability of LLMs. In FROG, we build the fuzzy reasoning tasks based on mathematical reasoning problems and explore common approaches designed to improve reasoning capabilities including math-specialized tuning, code-specialized tuning, and general alignment.

GQs are widely used to indicate proportions of predicate satisfaction in communication (Joshi et al., 2020) and benchmarks (Suhr et al., 2019; Apidianaki and Garí Soler, 2021). They also contribute as a major source to the deficiencies of NLP systems like NLI (Cui et al., 2022). Given their real-world prevalence, we employ GQs as a natural approach to introduce fuzzy information in FROG.

Existing methods for modeling fuzzy logic in natural language, as developed by Lee (2004) and Kapustin and Kapustin (2019), depend on pre-defined mapping functions to process fuzziness. Those mapping functions are mostly built from rule-based heuristics on limited data, or with simple distribution assumption (e.g. the gaussian distribution) and hard to be directly on real-world complex reasoning problems. In FROG, we rely on LLMs that are pretrained from large-scale real world text corpora to process the fuzziness of GQs and conduct complex math reasoning problems.

3 Benchmark Collection

Problems in FROG are collected from two math word problem datasets from the real world: GSM8K and MathQA. GSM8K includes grade school math word problems that can be largely solved with basic arithmetic operations. MathQA consists of multiple-choice GRE and GMAT-level math problems. In FROG, we include questions with percentage mentions, and Figure 1 displays an overview of the collection workflow. Specifically,

- **Step 1: Identifying Mathematical Questions with Percentage Mentions** - We begin by filtering the original questions to include only those that contain at least one percentage figure, of which the value is between 0% and 100%.
- **Step 2: Masking the Percentage Mention** - We obscure the specific target percentage mention by replacing it with a [MASK] token to construct a *Mask* question. If the original question contains multiple percentage mentions, each mention is masked out separately. We also employ two other masking strategies *Mislead* and *X%* where the target mention is substituted with a misleading quantifier (with the incorrectness pointed out in the FROG template) or *X%*.
- **Step 3: Searching for the Nearest Quantifiers** - The golden choice is selected by finding the closest GQ according to its average strength provided in QuRe (Li et al., 2023), a quantifier reasoning dataset with human-annotated quantifier strengths (i.e., *few* is the closest GQ to 10% in the running example) that do not include modifiers like particles that lead to semantic change.
- **Step 4: Constructing the FROG Task** - In FROG, we provide the question and the original answer to infer which GQ can be filled to represent the information masked out. This framework is driven by the acknowledgment that it is more practical to formulate fuzzy reasoning than directly solve mathematical problems with GQs. To carefully investigate the performance, we design the *easy* and *hard* mode of choices depending on the discriminability of misleading choices. The incorrect choices in FROG-Hard are the misleading top GQs in the previous step (e.g. *tiny amount*, *small amount*, *none* in the running example), while incorrect choices in FROG-Easy are randomly sampled from all misleading GQs. The original question, original answer and choices are then assembled through FROG templates. We refer to Appendix E for details.

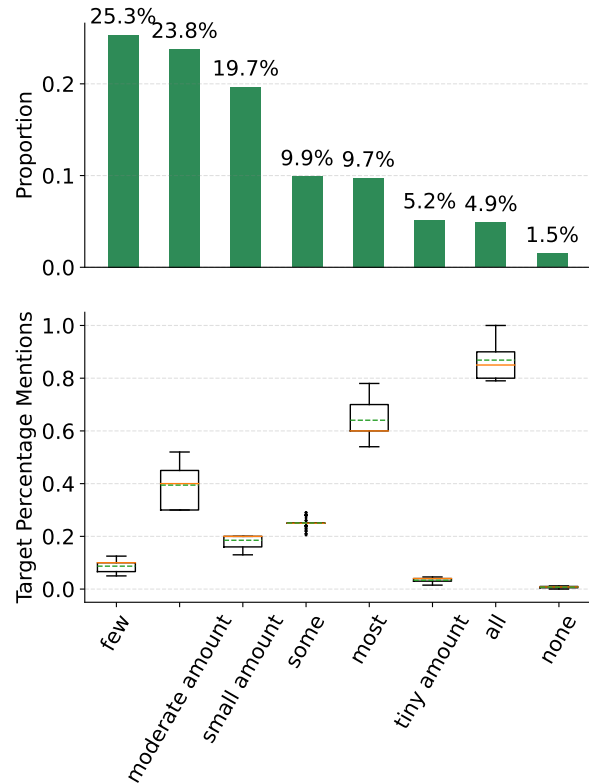


Figure 2: (Top) quantifier proportions in FROG. (Bottom) percentiles of target percentage mentions categorized by quantifiers. Green and orange lines represent the means and medians, respectively. The x-axis is shared between the two figures.

For MathQA data, the original multiple choice format is omitted in FROG since there is a negligible difference in performance between maintaining the original choice or providing the correct numeric answer, according to preliminary experiments.

Eventually, FROG includes 199 questions from GSM8K and 1,845 questions from MathQA. The average number of question tokens is 68.2, with each token separated by a space. And each question in FROG contains an average of 1.6 percentage mentions. The total number of quantifiers involved in FROG is 8, and the most common quantifiers used are *few* (25.3%), *moderate amount* (23.8%), and *small amount* (19.7%) (see top Figure 2 for details). The bottom of Figure 2 reveals the target percentage mentions mapped to each quantifier, e.g. the mean percentage value and 0.15 for *few* and around 0.4 for *moderate amount*. The mean strength difference between the golden quantifier and the incorrect quantifier that is closest to the target percentage mention is 0.105 in FROG-Hard.

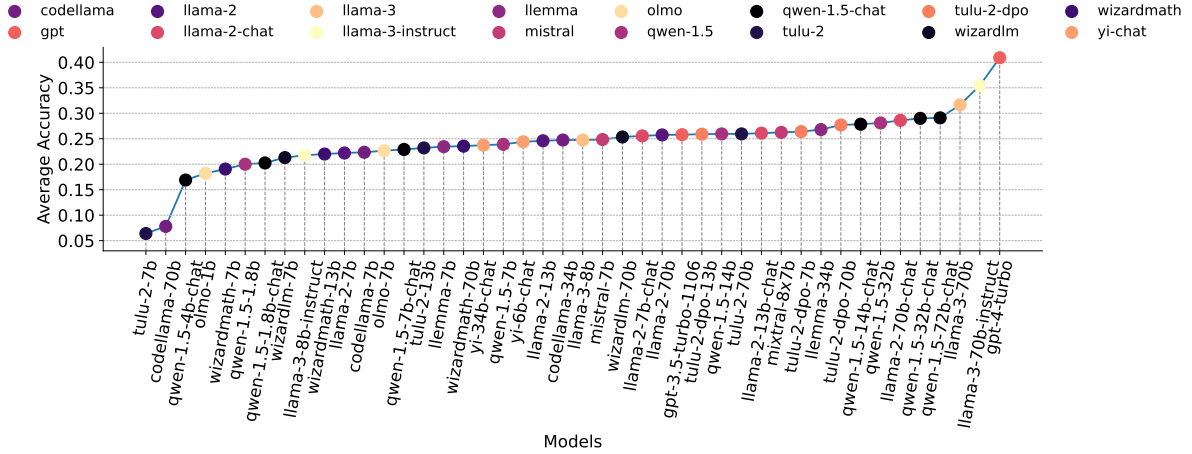


Figure 3: The average *Mask* accuracy in FROG-Easy and FROG-Hard of several LLMs sorting in ascending order. Dots with the same color belong to the same model family. Models with additional pretraining or instruction tuning do not necessarily perform better. We refer to Figure 5 and Figure 4 for more details.

4 Experiment

We evaluated several open-sourced LLMs, including Llama-2 (Touvron et al., 2023), CodeLlama (Rozière et al., 2024), Llemma (Azerbaiyev et al., 2024), Llama-3 (AI, 2024), Mistral (Jiang et al., 2023), Qwen-1.5 (Bai et al., 2023), Tulu-2 (Iverson et al., 2023), WizardLM (Xu et al., 2024), WizardMath (Luo et al., 2023) and Yi-Chat (Young et al., 2024) on our FROG benchmark. Specifically, we would like to investigate the following three research questions:

1. Are existing reasoning enhancement methods remain effective in FROG?
2. Can the scaling law be observed in FROG?
3. Is the mathematical reasoning strength transferable to FROG?

We employ the greedy decoding strategy with max tokens being 1,000, and temperature being 0.9 in experiments. The LLMs are instructed with task instructions (see Appendix F) and 5 demonstrations (Brown et al., 2020) with manually created chain-of-thought (Wei et al., 2022b) solutions to assist the reasoning procedure. The experiments are conducted on NVIDIA A100 80GB GPUs, each experiment can be finished within 2 hours.

Moreover, we investigate the sensitivity of different masking strategies. Specifically, we compare the performance between the *Mask* and *Mislead* or *X%* task by computing the Pearson and Spearman correlation of their accuracy. The results in Table 1

indicate strong positive correlations between the performance of *Mask* and *Mislead* or *X%*, meaning that LLMs are not sensitive to the masking strategy in FROG. We also do not observe strong correlation between accuracy and length of the generation. We choose the *Mask* task as the major task thereafter.

Task	Pearson	Spearman
Mislead	0.966/0.935	0.895/0.683
X%	0.988/0.980	0.958/0.889

Table 1: The Pearson and Spearman correlation between performance of different task and *Mask*. The two values corresponds to FROG-Easy and FROG-Hard, and all p values are smaller than 0.01. The performance of *Mislead* and *X%* are strongly correlated to *Mask*.

4.1 Overall Result

The result on FROG is displayed in Figure 3. In general, the accuracy of all models is around 0.05 and 0.45 (and mostly between 0.15 and 0.3), indicating that the fuzzy reasoning is a challenging task for the current LLMs. Moreover, models with small model sizes can demonstrate surprisingly strong performance in FROG compared to models much larger, e.g. Tulu-2-DPO-7B outperforms Tulu-2-70B, Llama-2-70B and WizardLM-70B.

4.2 Fine-grained Evaluation and Analysis

4.2.1 Q1: Are existing reasoning enhancement methods remain effective in FROG?

Instruction-tuning (Wei et al., 2022a; Ouyang et al., 2024) is demonstrated to further boost the abilities

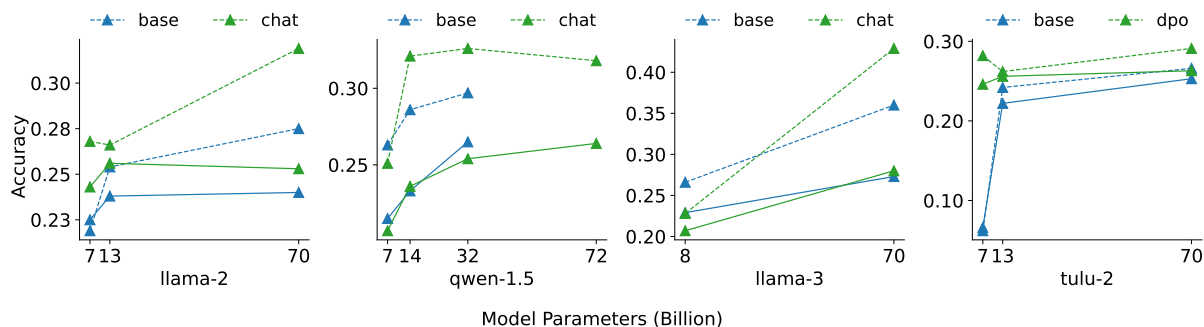


Figure 4: Comparison between different chat and base models of *Mask* on FROG. The solid and dashed lines represent the *hard* and *random* modes, respectively. Instruction-tuning does not necessarily improve the performance in FROG. The results of qwen-1.5-72b are full of punctuations and therefore omitted.

of LLMs (Zhang et al., 2023; Hu et al., 2024). It is employed to improve the usability and safety of LLM systems (e.g., the chat model Touvron et al. (2023); Wang et al. (2024a)), open-ended generations without sacrificing task-specific abilities (Iverson* et al., 2023) or mathematical reasoning abilities (Bai et al., 2023; Tang et al., 2024; Zhou and Zhao, 2024). Moreover, Zhang et al. (2024b) demonstrates consistent performance gain of LLMs on reasoning by instruction-tuning on code data, and the mathematical reasoning ability can be enhanced by introducing a continuous pre-training stage on mathematical tokens (Luo et al., 2023; Azerbayev et al., 2024).

We explore whether the effectiveness of continuous pretraining on math or code data, as well as general alignment tuning methods can be extended to fuzzy reasoning in FROG. In this regard, we selectively compare the performance of several models WizardLM, Qwen-1.5, Tulu-2, Llama-2, and Llama-3, and their instruction-tuning or aligned versions: WizardMath (based on WizardLM), CodeLlama (based on Llama-2), and Llemma (based on CodeLlama).

Math-specialized Tuning The results are demonstrated in Figure 5, where we observe that the accuracy of all LLMs are less than 30% and the mathematical continuous training does not bring universal benefits in FROG since WizardLM outperforms WizardMath in FROG. Besides, the scaling benefit of Llemma on the FROG-Easy (5.2% gain) does not comparably extend to the FROG-Hard (1.5%).

Code-specialized Tuning CodeLlama does not outperform Llama-2 in FROG, meaning continuous pretraining on code does not directly benefit fuzzy reasoning tasks. In fact, the domain shift from pro-

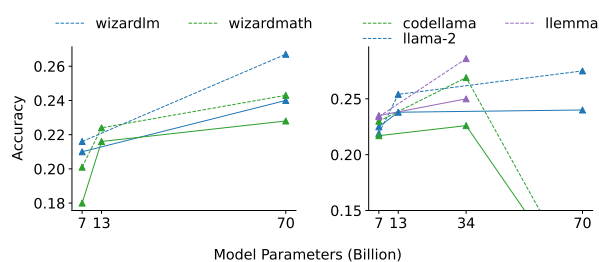


Figure 5: Impacts of continuous pretraining on mathematical data of LLMs on the performance of FROG. The solid and dashed lines represent FROG-Hard and FROG-Easy respectively. The result of CodeLlama (70B) is emitted for illustration due to its poor performance.

gramming to natural language largely impacts the performance. The largest CodeLlama (70B) evaluated directly generates code snippet most of the time, leading to the poor performance on FROG.

General Alignment The results are shown in Figure 4. In general, the performance on FROG-Easy (dashed lines) is better than FROG-Hard (solid lines) among all the models. Regarding the difference between the base models and their instruction-tuned versions, the benefit of instruction-tuning substantially diminishes from FROG-Easy to FROG-Hard in Llama-2, Qwen-1.5 and Llama-3. Lastly, the inverse scaling effect displays on Llama-2, Qwen-1.5 and Tulu-2. In FROG-Easy, the performance of 4 base models adhere to the scaling law, whereas the instruction-tuned models display inverse scaling effect except Llama-3. In FROG-Hard, Llama-2 demonstrates inverse scaling effect in both base and chat models. Conversely, Qwen-1.5 and Llama-3 consistently exhibit scaling phenomenon in both base and chat models. The Tulu-2 base model adheres to the scaling law, while the Tulu-2-DPO model display the inverse scaling effect.

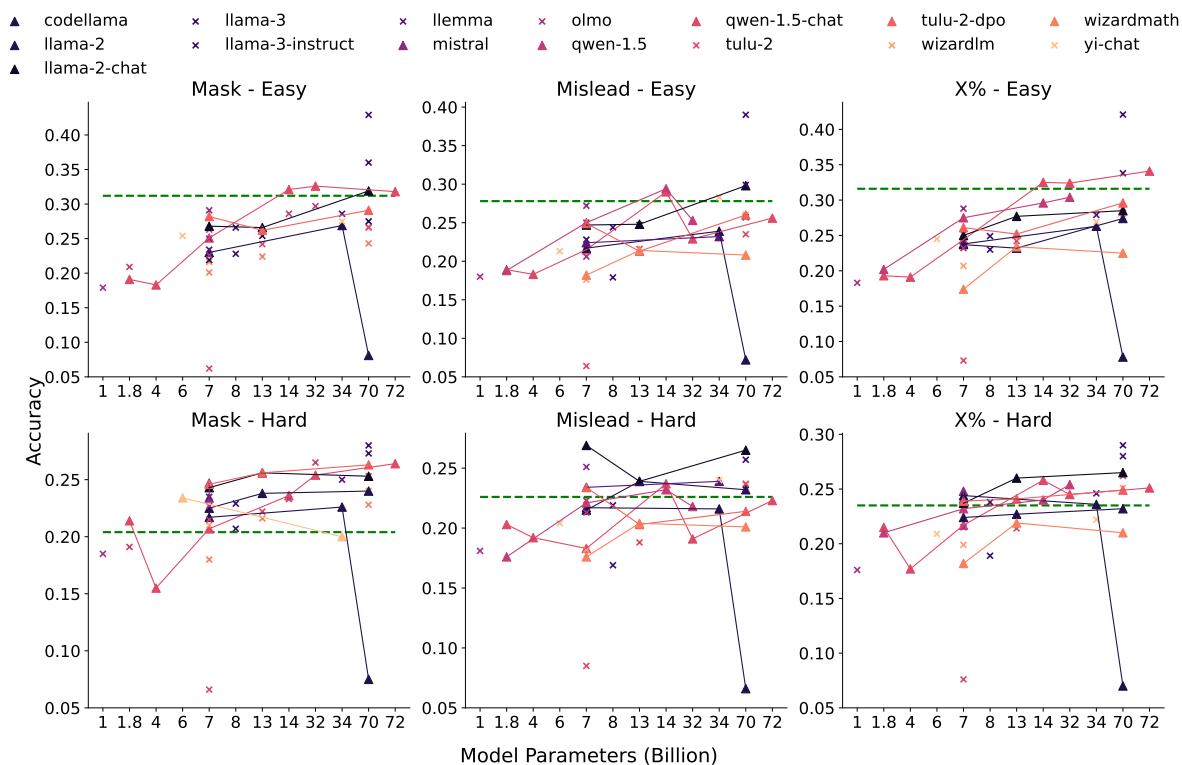


Figure 6: The performance of different LLMs on all FROG tasks with different masking strategies and difficulties. The solid lines represent models that demonstrate inverse scaling phenomenon, and crossings represent the performance of other models. The green line represents the performance of GPT-3.5-turbo-1106. More than 50% of the model families demonstrate the inverse scaling effect.

4.2.2 Q2: Can the scaling law be observed in FROG?

Scaling law is introduced in Kaplan et al. (2020) to suggest the phenomenon that LLMs can achieve enhanced task performance by scaling up model sizes. However, the scaling law does not hold universally. For example, the inverse scaling phenomenon can be observed when LLMs are instructed to choose which information can help to answer a question (McKenzie et al., 2023), which is similar to the design of FROG tasks. Here, we list the performance of all models evaluated on FROG in Figure 6. Each line highlights an observed inverse scaling effect (less than 1 point performance gain by scaling model sizes). It turns out that 8 out of the 15 model families evaluated demonstrate inverse scaling effect in FROG, crossing base models, continuous trained models and instruction-tuned models.

We further provide a case study of Qwen-1.5-Chat for its large number of open-resourced checkpoints and superior performance in mathematical reasoning. The results in Figure 7 show that the performance gap between FROG-Easy and FROG-

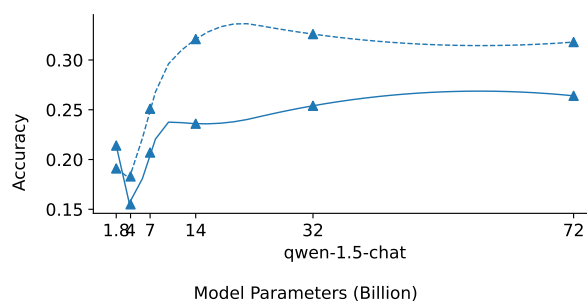


Figure 7: The accuracy of *Mask* of Qwen-1.5-Chat models, the real and dashed lines represent the hard and easy split, respectively.

Hard increases starting Qwen-1.5-4B-Chat, and the performance becomes saturated with a model of 7 billion parameters or larger. Moreover, the performance gain of scaling model parameters diminishes after 14 billion model parameters. Inverse scaling happens on the side of models smaller than 7 billion and larger than 32 billion parameters. Models with fewer than 14 billion parameters are very unstable, displaying poor performance (below 25%) and convoluted accuracy. Notably, the 4B model exhibits the poorest performance of all the models

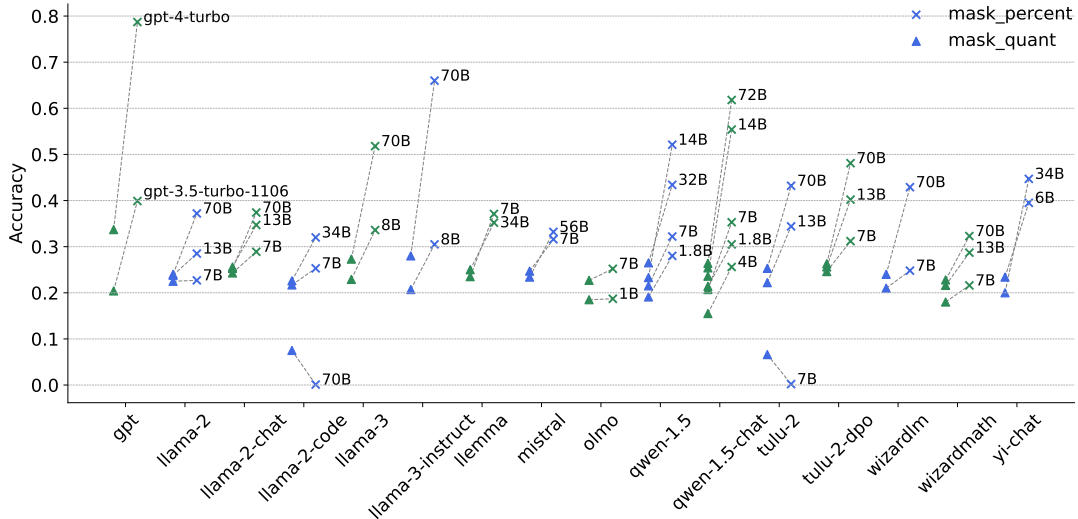


Figure 8: The performance comparison between the *mask_quant* and *mask_percent* on FRoG-Hard. Each dashed line connects the performance of the same model. LLMs with larger model sizes are more likely to receive larger performance degrade from *mask_percent* to *mask_quant* in FROG.

in FRoG-Easy and FRoG-Hard. Models with over 14 billion parameters, however, attain comparable performance in FRoG-Easy.

4.2.3 Q3: Is the mathematical reasoning strength transferrable to FROG?

Mathematical reasoning has become a key signal for reasoning capabilities of LLMs (Huang et al., 2024a). To proxy the fuzzy reasoning ability under the precise setting, we design a *mask_percent* baseline that substitutes the misleading choices into their corresponding mean average strengths in QuRe and the correct choice being the exact value of the target percentage mention. Therefore, the original *Mask* task (denoted as *mask_quant*) is transformed into figuring out the hidden percentage information in precise reasoning, with the predictions still reflect the preference over the quantifiers. For example, the *mask_percent* choices of question in Figure 1 are *A 0.4%, B 18.3%, C 10%, D 2.4%* under the hard mode.

The results are demonstrated in Figure 8, where GPT-4-turbo and Llama-3-Instruct are the best performed commercial or open-resourced model in both *mask_percent* and *mask_quant*. CodeLlama-70B and Tulu-2-7B achieve minimal accuracy in both precise reasoning (nearly 0%) and fuzzy reasoning (lower than 10%) tasks, which attributes to their programming or web content style outputs. The accuracy of *mask_percent* of a model is significantly higher than the *mask_quant* alternatives in most of the cases. But the advances

of *mask_percent* by scaling up model parameters are hard to transfer to the *mask_quant* task in FROG. In fact, among the 11 open-sourced models that achieve higher than 40% accuracy in the *mask_percent* setting, their accuracies in *mask_quant* are all lower than 30%. Take Tulu-2-DPO models for instance, even though the *mask_percent* performance improves by scaling the model size, the performance of those models drops significantly and the scaling effect shrinks drastically when shifted to the *mask_quant* task. Specifically, we observe that models with a larger number of parameters are more likely to receive a larger accuracy drop when shifted from *mask_percent* to *mask_quant*. The average accuracy drop of models smaller than 10 billion parameters is 6.8% compared to 16.9% of models of larger sizes, indicating that successfully solving a reverse mathematical (precise) reasoning problem does not necessarily reflect equivalent competences in solving FRoG problems with fuzziness introduced by GQs.

4.3 Case Study

Lastly, we are interested in the working mechanism that LLMs pose in conducting fuzzy reasoning in FROG. We randomly sample 50 results from GPT-4-turbo in FRoG-Hard, and demonstrate examples in Table 2. The mechanism of LLMs to solve fuzzy reasoning can be summarized as follows: firstly, the LLMs attempt to locate the target percentage value in three major patterns:

- (1) Precisely solving the target percentage mention.

Ex1: Kelly has had 3 pay cuts in her salary in the past 6 months. If the first pay cut was 8%, the second pay cut was 14%, and the third was [MASK]. What will be the percentage decrease, if the salary is decreased in a single shot? If the answer to the question is 35.12%, then please select the quantifier that is closest to the meaning of [MASK] from the following choices. [18%]

A. few B. some C. tiny amount D. small amount

Suppose that Kelly's initial salary is 100 units for simplicity. After the first pay cut of 8%, her salary becomes 92 units. The second pay cut of 14% reduces her salary further to $92 * (1 - 0.14) = 79.12$ units. Let the third pay cut be $x\%$. Therefore, her salary becomes $79.12 * (1 - x\%)$ after the third cut.

We need to find x such that the overall reduction is equivalent to a single reduction of 35.12%. This means the final salary should be $100 * (1 - 0.3512) = 64.88$ units.

Setting up the equation: $79.12 * (1 - x/100) = 64.88$.

Solving for x : $(1 - x/100) = 64.88 / 79.12 \approx 0.8202$, hence $x/100 = 1 - 0.8202 = 0.1798$ or $x = 17.98\%$.

Knowing the value of x , we can compare it to the options given:

- A few (typically refers to a small number, but greater than 'tiny amount')
- Some (could imply a moderate to considerable amount, usually more ambiguous)
- Tiny amount (implies very small, likely less than our calculated 17.98%)
- Small amount (suggests a non-trivial but not large percentage)

Between these choices, 'small amount' best describes a reduction of 17.98%, as it's significant but not very large or tiny. Therefore, **the best choice is D. small amount.**

Ex2: The average price of a decoration piece has increased over the years. If from 1990 to 1996, the price of the decoration piece increased by [MASK] and from 1996 to 2001 it increased by 20%, what is the price of the decoration piece in 2001 if the price in 1990 was \$11500? If the answer to the question is \$15594, then please select the quantifier that is closest to the meaning of [MASK] from the following choices. [13%]

A. some B. tiny amount C. few D. small amount

First, we need to calculate the price of the decoration piece in 1996 based on an unknown percentage increase from 1990. Let the percentage increase from 1990 to 1996 be $x\%$.

If the price in 1990 was \$11500, then the price in 1996 after $x\%$ increase would be $\$11500 * (1 + x/100)$.

From 1996 to 2001, the price increased further by 20%, so the price in 2001 would be: $\$11500 * (1 + x/100) * 1.2$

The final value is given as \$15594. So we have: $\$11500 * (1 + x/100) * 1.2 = \15594

Solving for x : $(1 + x/100) = \$15594 / (\$11500 * 1.2)$; $(1 + x/100) = \$15594 / \13800 ; $(1 + x/100) \approx 1.13$;

$x/100 = 0.13$ $x = 13\%$

So, the price increased by 13% from 1990 to 1996.

Given the choices of 'some,' 'tiny amount,' 'few,' and 'small amount,' a 13% increase is not negligible but is not very large either. Among the provided options, 'small amount' seems to be the closest match to a 13% increase.

Therefore, the answer is **D. small amount.**

Table 2: Sampled results in FRoG-Hard. The target percentage mention lies in the brackets, the correct answer is underscored and the prediction is bolded. The [explicit quantifier estimation stage](#) is highlighted.

(2) Reaching a percentage value that is close to the target percentage mention.

(3) Incomplete solution (e.g. unsolved equations).

After solving the target percentage mention, the LLMs select their preferred quantifiers through a quantifier estimation stage, which can be implicit. Such implicit reasoning procedure is also studied in precise math reasoning (Deng et al., 2023, 2024).

For example in Ex1 of Table 2, the model firstly solves the target percentage value with satisfying precision (17.98%), and then conduct a quantifier estimation stage before deciding the quantifier selection. Note that this quantifier selection stage is not always explicit (e.g. Ex2). We include more examples in Appendix C.

In general, the target percentage value can be located with satisfying precision (within 1% error) 52% of the time (still lower than the *mask_percent* setting where the target percentage value is provided as a candidate choice), with only 61% of

them lead to the final correctness of the problems. It indicates the complexity of conducting fuzzy reasoning even with the target percentage value given.

Moreover, 15 of the 50 predictions show explicit quantifier estimation by comparing or estimating the strengths of quantifiers, but only 5 of them lead to the correct quantifier predictions, meaning that precisely solving the target percentage value is insufficient to conduct fuzzy reasoning. Meanwhile, there are 15 cases where models select the correct quantifier without explicit quantifier estimations, indicating that models may rely on implicit mechanisms in conducting fuzzy reasoning.

Note that even though models within the same model family but with different number of parameters can figure out the target percentage mention correctly or close enough, their interpretation of quantifiers can make a difference in the final prediction, we refer to Appendix D for examples.

5 Conclusion

The fuzzy reasoning ability is an under-explored direction of the reasoning ability of LLMs. To measure the fuzzy reasoning ability of LLMs, we collect a fuzzy reasoning benchmark FROG that is based on generalized quantifiers. The experimental results show that fuzzy reasoning remains challenging for current LLMs, and an inverse scaling effect is observed on the performance of FROG. Besides, prevailing reasoning enhancement approaches including continuous pretraining, instruction tuning and general alignment may not stay effective on fuzzy reasoning of FROG. Lastly, LLMs can demonstrate diverse behaviors in fuzzy reasoning.

Acknowledgments

The authors would like to thank the anonymous reviewers for their suggestions and feedback on the work. This work was partially funded by the National Natural Science Foundation of China (62476168), Qingyuan Research Project. The views contained in this article are those of the authors and not of the funding agency.

Limitations

In this work, we collect a fuzzy reasoning dataset FROG to evaluate the fuzzy reasoning abilities of several existing LLMs. We are aware that even though the problems in FROG originate from real-world math word problems, the new question created may not naturally occur, and the designed masking-based reasoning protocol is not identical to the real-world reasoning procedure where the vague information is processed directly. We also note that GQ-based fuzzy reasoning is only a subset of the entire family of natural language fuzzy reasoning, and the scope of GQs is broader than the ones being studied in this work.

References

Meta AI. 2024. Introducing meta llama 3: The most capable openly available llm to date.

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. [MathQA: Towards interpretable math word problem solving with operation-based formalisms](#). 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

2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2023. [Learning from mistakes makes llm better reasoner](#). *arXiv preprint arXiv:2310.20689*.

Marianna Apidianaki and Aina Garí Soler. 2021. [ALL dolphins are intelligent and SOME are friendly: Probing BERT for nouns’ semantic properties and their prototypicality](#). In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 79–94, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen Marcus McAleer, Albert Q. Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2024. [Llemma: An open language model for mathematics](#). In *The Twelfth International Conference on Learning Representations*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Sheng-guang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Chen Bowen, Rune Sætre, and Yusuke Miyao. 2024. [A comprehensive evaluation of inductive reasoning capabilities and problem solving in large language models](#). In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 323–339, St. Julian’s, Malta. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Jiaao Chen, Xiaoman Pan, Dian Yu, Kaiqiang Song, Xiaoyang Wang, Dong Yu, and Jianshu Chen. 2023a.

- Skills-in-context prompting: Unlocking compositionality in large language models. *arXiv preprint arXiv:2308.00304*.
- Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. 2023b. [Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks](#). *Transactions on Machine Learning Research*.
- Yew Ken Chia, Guizhen Chen, Luu Anh Tuan, Soujanya Poria, and Lidong Bing. 2023. Contrastive chain-of-thought prompting. *arXiv preprint arXiv:2311.09277*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Ruixiang Cui, Daniel Hershcovich, and Anders Søgaard. 2022. [Generalized quantifiers as a source of error in multilingual NLU benchmarks](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4875–4893, Seattle, United States. Association for Computational Linguistics.
- DeepSeek-AI, :, Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiu Shi Du, Zhe Fu, Huazuo Gao, Kaige Gao, Wenjun Gao, Ruiqi Ge, Kang Guan, Daya Guo, Jianzhong Guo, Guangbo Hao, Zhewen Hao, Ying He, Wenjie Hu, Panpan Huang, Erhang Li, Guowei Li, Jiashi Li, Yao Li, Y. K. Li, Wenfeng Liang, Fangyun Lin, A. X. Liu, Bo Liu, Wen Liu, Xiaodong Liu, Xin Liu, Yiyuan Liu, Haoyu Lu, Shanghao Lu, Fuli Luo, Shirong Ma, Xiaotao Nie, Tian Pei, Yishi Piao, Junjie Qiu, Hui Qu, Tongzheng Ren, Zehui Ren, Chong Ruan, Zhangli Sha, Zhihong Shao, Junxiao Song, Xuecheng Su, Jingxiang Sun, Yaofeng Sun, Minghui Tang, Bingxuan Wang, Peiyi Wang, Shiyu Wang, Yaohui Wang, Yongji Wang, Tong Wu, Y. Wu, Xin Xie, Zhenda Xie, Ziwei Xie, Yiliang Xiong, Hanwei Xu, R. X. Xu, Yanhong Xu, Dejian Yang, Yuxiang You, Shuiping Yu, Xingkai Yu, B. Zhang, Haowei Zhang, Lecong Zhang, Liyue Zhang, Mingchuan Zhang, Minghua Zhang, Wentao Zhang, Yichao Zhang, Chenggang Zhao, Yao Zhao, Shangyan Zhou, Shunfeng Zhou, Qihao Zhu, and Yuheng Zou. 2024. [Deepseek llm: Scaling open-source language models with longtermism](#). *Preprint*, arXiv:2401.02954.
- Yuntian Deng, Yejin Choi, and Stuart Shieber. 2024. [From explicit cot to implicit cot: Learning to internalize cot step by step](#). *Preprint*, arXiv:2405.14838.
- Yuntian Deng, Kiran Prasad, Roland Fernandez, Paul Smolensky, Vishrav Chaudhary, and Stuart Shieber. 2023. [Implicit chain of thought reasoning via knowledge distillation](#). *Preprint*, arXiv:2311.01460.
- Usef Faghihi, Serge Robert, Pierre Poirier, and Youssef Barkaoui. 2020. From association to reasoning, an alternative to pearls’ causal reasoning. In *The Thirty-Third International Flairs Conference*.
- Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Pal: Program-aided language models. In *International Conference on Machine Learning*, pages 10764–10799. PMLR.
- Zhibin Gou, Zhihong Shao, Yeyun Gong, yelong shen, Yujia Yang, Nan Duan, and Weizhu Chen. 2024a. [CRITIC: Large language models can self-correct with tool-interactive critiquing](#). In *The Twelfth International Conference on Learning Representations*.
- Zhibin Gou, Zhihong Shao, Yeyun Gong, yelong shen, Yujia Yang, Minlie Huang, Nan Duan, and Weizhu Chen. 2024b. [ToRA: A tool-integrated reasoning agent for mathematical problem solving](#). In *The Twelfth International Conference on Learning Representations*.
- 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. *NeurIPS*.
- Hanxu Hu, Pinzhen Chen, and Edoardo M Ponti. 2024. Fine-tuning large language models with sequential instructions. *arXiv preprint arXiv:2403.07794*.
- Jie Huang and Kevin Chen-Chuan Chang. 2023. [Towards reasoning in large language models: A survey](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1049–1065, Toronto, Canada. Association for Computational Linguistics.
- Yiming Huang, Xiao Liu, Yeyun Gong, Zhibin Gou, Yelong Shen, Nan Duan, and Weizhu Chen. 2024a. [Key-point-driven data synthesis with its enhancement on mathematical reasoning](#). *arXiv preprint arXiv:2403.02333*.
- Yinya Huang, Xiaohan Lin, Zhengying Liu, Qingxing Cao, Huajian Xin, Haiming Wang, Zhenguo Li, Linqi Song, and Xiaodan Liang. 2024b. [MUSTARD: Mastering uniform synthesis of theorem and proof data](#). In *The Twelfth International Conference on Learning Representations*.
- Hamish Ivison*, Yizhong Wang*, Valentina Pyatkin, Nathan Lambert, Matthew Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023. [Camels in a changing climate: Enhancing lm adaptation with tulu 2](#). *arXiv preprint*.
- 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*.

- Pratik Joshi, Somak Aditya, Aalok Sathe, and Monojit Choudhury. 2020. [TaxiNLI: Taking a ride up the NLU hill](#). In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 41–55, Online. Association for Computational Linguistics.
- Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. [Scaling laws for neural language models](#). *Preprint*, arXiv:2001.08361.
- Pavlo Kapustin and Michael Kapustin. 2019. [Modeling language constructs with fuzzy sets: some approaches, examples and interpretations](#). In *Proceedings of the 13th International Conference on Computational Semantics - Student Papers*, pages 24–33, Gothenburg, Sweden. Association for Computational Linguistics.
- Kwang Hyung Lee. 2004. *First course on fuzzy theory and applications*, volume 27. Springer Science & Business Media.
- Aitor Lewkowycz, Anders Johan Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Venkatesh Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. 2022. [Solving quantitative reasoning problems with language models](#). In *Advances in Neural Information Processing Systems*.
- Haoran Li, Qingxiu Dong, Zhengyang Tang, Chaojun Wang, Xingxing Zhang, Haoyang Huang, Shaohan Huang, Xiaolong Huang, Zeqiang Huang, Dongdong Zhang, et al. 2024. [Synthetic data \(almost\) from scratch: Generalized instruction tuning for language models](#). *arXiv preprint arXiv:2402.13064*.
- Yiyuan Li, Rakesh Menon, Sayan Ghosh, and Shashank Srivastava. 2023. [Pragmatic reasoning unlocks quantifier semantics for foundation models](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 573–591, Singapore. Association for Computational Linguistics.
- Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jianguang Lou, Chongyang Tao, Xiubo Geng, Qingwei Lin, Shifeng Chen, and Dongmei Zhang. 2023. [Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct](#). *Preprint*, arXiv:2308.09583.
- Ian R. McKenzie, Alexander Lyzhov, Michael Martin Pieler, Alicia Parrish, Aaron Mueller, Ameya Prabhu, Euan McLean, Xudong Shen, Joe Cavanagh, Andrew George Gritsevskiy, Derik Kauffman, Aaron T. Kirtland, Zhengping Zhou, Yuhui Zhang, Sicong Huang, Daniel Wurgaft, Max Weiss, Alexis Ross, Gabriel Recchia, Alisa Liu, Jiacheng Liu, Tom Tseng, Tomasz Korbak, Najoung Kim, Samuel R. Bowman, and Ethan Perez. 2023. [Inverse scaling: When bigger isn't better](#). *Transactions on Machine Learning Research*. Featured Certification.
- Swaroop Mishra, Matthew Finlayson, Pan Lu, Leonard Tang, Sean Welleck, Chitta Baral, Tanmay Rajpurohit, Oyvind Tafjord, Ashish Sabharwal, Peter Clark, and Ashwin Kalyan. 2022. [LILA: A unified benchmark for mathematical reasoning](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5807–5832, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Andrzej Mostowski. 1957. [On a generalization of quantifiers](#). *Fundamenta Mathematicae*, 44(1):12–36.
- Vilém Novák. 2015. *Fuzzy Natural Logic: Towards Mathematical Logic of Human Reasoning*, pages 137–165. Springer International Publishing, Cham.
- Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2024. [Training language models to follow instructions with human feedback](#). In *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*, Red Hook, NY, USA. Curran Associates Inc.
- Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. 2024. [Openwebmath: An open dataset of high-quality mathematical web text](#). In *The Twelfth International Conference on Learning Representations*.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. [Direct preference optimization: Your language model is secretly a reward model](#). In *Thirty-seventh Conference on Neural Information Processing Systems*.
- Sonia Ramotowska, Julia M. Haaf, Leendert van Maanen, and Jakub Szymanik. 2024. [Most quantifiers have many meanings](#). *Psychonomic bulletin & review*.
- Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. 2024. [Code llama: Open foundation models for code](#). *Preprint*, arXiv:2308.12950.
- Amir Saki and Usef Faghihi. 2022. [A fundamental probabilistic fuzzy logic framework suitable for causal reasoning](#). *Preprint*, arXiv:2205.15016.
- Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2019. [A corpus for](#)

- reasoning about natural language grounded in photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428, Florence, Italy. Association for Computational Linguistics.
- Zhengyang Tang, Xingxing Zhang, Benyou Wan, and Furu Wei. 2024. Mathsacle: Scaling instruction tuning for mathematical reasoning. *arXiv preprint arXiv:2403.02884*.
- Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. *Galactica: A large language model for science*. Preprint, arXiv:2211.09085.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. *Llama 2: Open foundation and finetuned chat models*. Preprint, arXiv:2307.09288.
- Jiahao Wang, Bolin Zhang, Qianlong Du, Jiajun Zhang, and Dianhui Chu. 2024a. A survey on data selection for llm instruction tuning. *arXiv preprint arXiv:2402.05123*.
- Junlin Wang, Siddhartha Jain, Dejjiao Zhang, Baishakhi Ray, Varun Kumar, and Ben Athiwaratkun. 2024b. Reasoning in token economies: Budget-aware evaluation of llm reasoning strategies. *arXiv preprint arXiv:2406.06461*.
- Ling Wang, Qian Ma, and Jianyao Meng. 2019. *Incremental fuzzy association rule mining for classification and regression*. *IEEE Access*, 7:121095–121110.
- Jason Wei, Maarten Bosma, Vincent 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*. In *International Conference on Learning Representations*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le, and Denny Zhou. 2022b. *Chain of thought prompting elicits reasoning in large language models*. In *Advances in Neural Information Processing Systems*.
- Shijie Xia, Xuefeng Li, Yixin Liu, Tongshuang Wu, and Pengfei Liu. 2024. Evaluating mathematical reasoning beyond accuracy. *arXiv preprint arXiv:2404.05692*.
- Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. 2024. *WizardLM: Empowering large pre-trained language models to follow complex instructions*. In *The Twelfth International Conference on Learning Representations*.
- Ronald R. Yager and Lotfi A. Zadeh. 1992. *An Introduction to Fuzzy Logic Applications in Intelligent Systems*. Kluwer Academic Publishers, USA.
- Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, et al. 2024. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*.
- Fei Yu, Hongbo Zhang, Prayag Tiwari, and Benyou Wang. 2023. Natural language reasoning, a survey. *ACM Computing Surveys*.
- Longhui Yu, Weisen Jiang, Han Shi, Jincheng YU, Zhengying Liu, Yu Zhang, James Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2024. *Metamath: Bootstrap your own mathematical questions for large language models*. In *The Twelfth International Conference on Learning Representations*.
- Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao Huang, Huan Sun, Yu Su, and Wenhui Chen. 2024. *MAMmoTH: Building math generalist models through hybrid instruction tuning*. In *The Twelfth International Conference on Learning Representations*.
- Lotfi A. Zadeh. 1968. *Probability measures of fuzzy events*. *Journal of Mathematical Analysis and Applications*, 23:421–427.
- Beichen Zhang, Kun Zhou, Xilin Wei, Xin Zhao, Jing Sha, Shijin Wang, and Ji-Rong Wen. 2024a. Evaluating and improving tool-augmented computation-intensive math reasoning. *Advances in Neural Information Processing Systems*, 36.
- Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.
- Xinlu Zhang, Zhiyu Zoey Chen, Xi Ye, Xianjun Yang, Lichang Chen, William Yang Wang, and Linda Ruth Petzold. 2024b. *Unveiling the impact of coding data instruction fine-tuning on large language models reasoning*. Preprint, arXiv:2405.20535.

Chuanyang Zheng, Zhengying Liu, Enze Xie, Zhenguo Li, and Yu Li. 2023. Progressive-hint prompting improves reasoning in large language models. *arXiv preprint arXiv:2304.09797*.

Aojun Zhou, Ke Wang, Zimu Lu, Weikang Shi, Sichun Luo, Zipeng Qin, Shaoqing Lu, Anya Jia, Linqi Song, Mingjie Zhan, and Hongsheng Li. 2024. Solving challenging math word problems using GPT-4 code interpreter with code-based self-verification. In *The Twelfth International Conference on Learning Representations*.

Yongwei Zhou and Tiejun Zhao. 2024. Dual instruction tuning with large language models for mathematical reasoning. *arXiv preprint arXiv:2403.18295*.

Kaijie Zhu, Jiaao Chen, Jindong Wang, Neil Zhenqiang Gong, Diyi Yang, and Xing Xie. 2024. Dyval: Dynamic evaluation of large language models for reasoning tasks. In *The Twelfth International Conference on Learning Representations*.

A Quantifier Semantics in FROG

To further measure the difficulty of FROG questions, we introduce the relative distance between the target percentage value and the GQ choices. Specifically, given a question, the value of target percentage mention p , the mean strength of the golden GQ choice q_g and the mean strength of the incorrect GQ choice q_m that is closest to p , the relative distance RelDist is calculated by

$$\text{RelDist} = \frac{\text{abs}(q_g - p)}{\text{abs}(q_g - p) + \text{abs}(q_m - p)}$$

where $\text{abs}(\ast)$ represents the operation of computing the absolute value. Here, RelDist indicates the proportional closeness of p to q_g and lower is better. The average RelDist is 0.3 in FROG. 21.3% questions have RelDist smaller than 0.2, 63.8% have RelDist smaller than 0.4, and all questions have RelDist smaller than 0.5. The average RelDist by generalized quantifiers is listed in Table 3.

B FROG Performance

The performance of FROG on three masking strategies is listed in Table 4, the best performing model, GPT-4-turbo has below 50% accuracy across different settings. The results of inverse scaling effect are summarized in Table 5 where one check represents an inverse scaling effect observed.

C Additional Case Studies

We list additional examples from GPT-4-turbo on FROG-Hardin Table 6, where Ex1, Ex2 and Ex3

QUANTIFIER	RELDIST
all	0.276
most	0.249
moderate amount	0.311
some	0.276
small amount	0.364
few	0.293
tiny amount	0.255
none	0.211

Table 3: The average RelDist by generalized quantifiers in FROG. All quantifiers have RelDist lower than 0.4.

represents the three primary fuzzy reasoning strategies in Section 4.3 respectively. For example, in Ex2, the model solves the target percentage mention with 30% instead of 20%, but still select the correct quantifier. And in Ex3, the model skips the step of dividing 70 by 127 and directly estimate the GQ preference from it.

D Quantifier Understanding Across Scale

We demonstrate examples from Qwen-1.5-Chat in FROG-Hardwhere models of different number of parameters hold different understanding of GQ semantics in Table 7. Take Ex1 for example, The 7B, 14B and 72B version of Qwen-1.5-Chat all compute the target target mention 0.5 correctly, but reaching to different GQ preferences (*small amount, moderate amount and some*) regarding interpreting the target percentage value. We list the exploration of aligning model behavior to specific quantifier interpretation as future work.

E FROG Templates

We list the FROG templates employed in Table 8.

F Instruction

We list the FROG instruction employed in Table 9.

MODEL	#PARAM.	MASK	MISLEAD	X %
GPT-4-turbo	-	33.7 / 48.1	34.7 / 44.0	37.5 / 49.8
GPT-3.5-turbo-1106	-	20.4 / 31.2	22.6 / 27.8	23.5 / 31.6
Llama-2-7b	7B	22.5 / 21.9	21.5 / 22.8	22.4 / 23.7
Llama-2-7b-Chat	7B	24.3 / 26.8	26.9 / 24.7	23.7 / 25.0
Llama-2-13b	13B	23.8 / 25.4	23.9 / 24.7	22.7 / 23.2
Llama-2-13b-Chat	13B	25.6 / 26.6	23.9 / 24.8	26.0 / 27.7
Llama-2-70b	70B	24.0 / 27.5	23.2 / 25.7	23.2 / 27.4
Llama-2-70b-Chat	70B	25.3 / 31.9	26.5 / 29.8	26.5 / 28.5
CodeLlama-7b	7B	21.7 / 23.0	21.7 / 21.7	24.4 / 23.8
CodeLlama-34b	34B	22.6 / 26.9	21.6 / 23.9	23.6 / 26.3
CodeLlama-70b	70B	7.5 / 8.1	6.6 / 7.2	7.0 / 7.8
Llama-3-8b	8B	22.9 / 26.6	21.9 / 24.4	23.8 / 24.9
Llama-3-8b-Instruct	8B	20.7 / 22.8	16.9 / 17.9	18.9 / 23.0
Llama-3-70b	70B	27.3 / 36.0	23.2 / 29.9	28.0 / 33.8
Llama-3-70b-Instruct	70B	28.0 / 42.9	25.7 / 39.0	29.0 / 42.1
Llemma-7b	7B	23.5 / 23.4	23.4 / 22.4	23.6 / 24.2
Llemma-34b	34B	25.0 / 28.6	23.9 / 23.2	24.6 / 27.9
Mistral-7b	7B	24.7 / 25.0	22.3 / 25.0	24.1 / 25.2
Mixtral-8x7b	56B	23.4 / 29.1	25.1 / 27.2	24.8 / 28.8
Olmo-1b	1B	18.5 / 17.9	18.1 / 18.0	17.6 / 18.3
Olmo-7b	7B	22.7 / 22.6	21.3 / 20.6	21.5 / 23.2
Qwen-1.5-1.8b	1.8B	19.1 / 20.9	17.6 / 18.8	21.0 / 20.2
Qwen-1.5-1.8b-Chat	1.8B	21.4 / 19.1	20.3 / 18.9	21.5 / 19.3
Qwen-1.5-4b-Chat	4B	15.5 / 18.3	19.2 / 18.3	17.7 / 19.1
Qwen-1.5-7b	7B	21.5 / 26.3	22.1 / 25.0	23.2 / 27.5
Qwen-1.5-7b-Chat	7B	20.7 / 25.1	18.3 / 21.5	21.7 / 24.4
Qwen-1.5-14b	14B	23.3 / 28.6	23.2 / 29.4	24.0 / 29.6
Qwen-1.5-14b-Chat	14B	23.6 / 32.1	23.7 / 29.0	25.8 / 32.5
Qwen-1.5-32b	32B	26.5 / 29.7	21.8 / 25.3	25.4 / 30.4
Qwen-1.5-32b-Chat	32B	25.4 / 32.6	19.1 / 22.9	24.5 / 32.4
Qwen-1.5-72b-Chat	72B	26.4 / 31.8	22.3 / 25.6	25.1 / 34.1
Tulu-2-7b	7B	6.6 / 6.2	8.5 / 6.4	7.6 / 7.3
Tulu-2-DPO-7b	7B	24.6 / 28.2	23.4 / 24.9	23.9 / 26.1
Tulu-2-13b	13B	22.2 / 24.2	18.8 / 21.6	21.4 / 24.2
Tulu-2-DPO-13b	13B	25.6 / 26.2	20.3 / 21.3	24.1 / 25.2
Tulu-2-70b	70B	25.3 / 26.6	23.7 / 23.5	26.2 / 28.9
Tulu-2-DPO-70b	70B	26.3 / 29.1	21.4 / 26.0	24.9 / 29.6
WizardLM-7b*	7B	21.0 / 21.6	18.2 / 17.6	19.9 / 20.7
WizardMath-7b*	7B	18.0 / 20.1	17.6 / 18.2	18.2 / 17.4
WizardMath-13b*	13B	21.6 / 22.4	20.4 / 21.4	21.9 / 23.4
WizardLM-70b*	70B	24.0 / 26.7	23.6 / 26.0	25.1 / 27.4
WizardMath-70b*	70B	22.8 / 24.3	20.1 / 20.8	21.0 / 22.5
Yi-6b-Chat	6B	23.4 / 25.4	20.4 / 21.3	20.9 / 24.5
Yi-34b-Chat	34B	20.0 / 27.5	24.0 / 28.3	22.2 / 26.9

Table 4: Performance on FROG. * number of demonstrations is reduced to 3 avoid outputs truncated by length limits.

MODEL	BACKBONE	[MASK]	MISLEAD	X%
Tulu-2	Llama			
Tulu-2-DPO	Llama	✓✓	✓✓	✓✓
WizardLM	Llama			
WizardMath	Llama	✓	✓✓	✓✓
Llama-2	Llama-2	✓	✓	✓✓
Llama-2-Chat	Llama-2	✓	✓✓	✓✓
CodeLlama	Llama-2	✓✓	✓✓	✓✓
Llemma	Llama-2		✓✓	
Llama-3	Llama-3			
Llama-3-Instruct	Llama-3			
Mistral	Mistral	✓		✓
Olmo	Olmo			
Qwen-1.5	Qwen-1.5		✓✓	✓
Qwen-1.5-Chat	Qwen-1.5	✓✓	✓✓	✓✓
Yi-Chat	Yi	✓		

Table 5: Inverse scaling checkbox, the two checks represent inverse scaling effect (less than 1 point performance gain by scaling model sizes) observed on FRoG-Easy (left) and FRoG-Hard (right) respectively.

Ex1: From the sale of sleeping bags, a retailer made a gross profit of [MASK] of the wholesale cost. If each sleeping bag was sold for \$28, what was the wholesale cost per bag?
 If the answer to the question is 24.78, then please select the quantifier that is closest to the meaning of [MASK] from the following choices. [13%]
 A. some B. small amount C. few D. tiny amount

Suppose the gross profit percentage is x and the wholesale cost is W per bag. Then the selling price (SP) per bag is \$28, and the actual relation can be established using: $SP = W + x\%$ of W $28 = W + (x/100) * W$ $28 = W(1 + x/100)$ $W = 28 / (1 + x/100)$
 Given that W is \$24.78, $24.78 = 28 / (1 + x/100)$ $1 + x/100 = 28 / 24.78$ $x/100 = (28 / 24.78) - 1$ $x/100 = 1.13 - 1$ $x/100 = 0.13$ $x = 13$
 Therefore, the gross profit percentage is 13%. This percentage is a moderate amount, not minimal, nor very large.
 Checking the choices, none represent precisely '13%', but thinking about its relative value:
 - 'some' generally suggests a moderate, indefinite amount
 - 'small amount', 'few', and 'tiny amount' suggest smaller proportions
 Thus, among all the choices, 'some' is the best fit for representing a gross profit of 13% of the wholesale cost.

Ex2: A shopkeeper has 280 kg of apples. He sells 30% of these at 20% profit and the remaining 60% at [MASK] profit. Find his % profit on total. If the answer to the question is 24%, then please select the quantifier that is closest to the meaning of [MASK] from the following choices. [20%]
 A. some B. moderate amount C. small amount D. few

The shopkeeper has 280 kg of apples. 30% of these apples sold at 20% profit means $0.3 * 280 = 84$ kg sold at 20% profit. The remaining 60% sold at an unknown percentage, x , of profit is $0.6 * 280 = 168$ kg.
 To find the total profit percentage on the whole 280 kg:
 1. Calculation for profit from the first 84 kg: Profit = 20% of 84 kg = $0.2 * 84 = 16.8$ kg equivalent profit.
 2. Calculation for profit from the remaining 168 kg: Profit = $x\%$ of 168 kg = $0.01x * 168 = 1.68x$ kg equivalent profit.
 Now, total profit from 280 kg = $(16.8 + 1.68x)$ kg.
 Total profit percentage = $(\text{Total profit} / \text{Total kg}) * 100 = [(16.8 + 1.68x) / 280] * 100 = (0.06 + 0.006x) * 100 = 6 + 0.6x$.
 Given the problem, the total profit percentage is 24%: $6 + 0.6x = 24$ $0.6x = 24 - 6$ $0.6x = 18$ $x = 18 / 0.6$ $x = 30\%$.
 So, the remaining 60% of apples were sold at 30% profit. Looking at the choices given for the interpretation of [MASK], the closest to a 30% profit level among the options provided is "moderate amount", considering it reflects a sizable yet not excessively high-profit margin.

Ex3: A candidate appearing for an examination has to secure [MASK] marks to pass paper i. But he secured only 45 marks and failed by 25 marks. What is the maximum mark for paper i? If the answer to the question is 127, then please select the quantifier that is closest to the meaning of [MASK] from the following choices. [55%]
 A. moderate amount B. some C. all D. most

The candidate secured 45 marks and failed by 25 marks. This means the passing mark for paper I is 45 marks + 25 marks = 70 marks.
 Since the maximum mark for paper I is given to be 127, the passing mark (70 marks) out of the maximum (127 marks) represents more than 50% but significantly less than 100%. This suggests a quantifier that indicates a majority but not the entirety or something overly general.
 Among the choices:
 A. moderate amount - This generally indicates a reasonable or average proportion, not specifically majority.
 B. some - Generally refers to an unspecified quantity, not necessarily a majority.
 C. all - This would indicate the entirety, which is not suitable here as 70 is less than 127.
 D. most - This generally refers to the majority or the largest part but not the entirety, fitting the criterion where 70 is more than half of 127 but not all of it.
 Therefore, 'most' is the quantifier that is closest to the meaning required for the passing marks in the context given. ####
D

Table 6: Extra sampled results from GPT-4-turbo in FROG, where explicit quantifier estimation is highlighted.

Ex1 Mary can do a piece of work in 12 days. rosy is [MASK] more efficient than mary. the number of days taken by rosy to do the same piece of work is? If the answer to the question is 8, then please select the quantifier that is closest to the meaning of [MASK] from the following choices. [50%]

A. moderate amount B. some C. most D. small amount

[Qwen-1.5-72B-Chat] If Mary takes 12 days to complete the work, Rosy, who is more efficient, will take less time. Let's say Mary's efficiency is 1 unit of work per day, then Rosy's efficiency is Mary's plus [MASK] units. If Rosy is 'x' times more efficient than Mary, her efficiency would be (1+x) units per day.

Since the work is the same for both, we can equate the efficiency in terms of time:

Mary: 1 unit/day * 12 days = 12 units

Rosy: (1+x) units/day * d days = 12 units

Solving for d, we have $12 / (1 + x) = d$

Given the answer is 8 days for d, we can plug this value back into the equation to find x:

$$12 / (1 + x) = 8$$

$$1 + x = 12 / 8$$

$$1 + x = 1.5$$

$$x = 0.5$$

Since 0.5 represents a "small amount" more efficiency, the answer is **D. small amount. #### D**

[Qwen-1.5-14B-Chat] Suppose Mary's work efficiency is 'M' days' work. Then, Rosy's efficiency is 'M + [MASK]' days' work. Given, Rosy can complete the work in 'X' days.

So, $(M + [MASK]) * X = M * 12$

Given X = 8 days, we can find what [MASK] represents:

$$12M / X = M + [MASK]$$

$$12 / 8 = 1 + [MASK]$$

$$[MASK] = 12 / 8 - 1$$

$$[MASK] = 3/2 - 1$$

$$[MASK] = 1/2$$

This means Rosy is half as efficient as Mary, or in other words, she takes twice the time. This corresponds to a "moderate amount" of difference because, compared to 'most' or a large amount, which would be a huge difference, a small amount wouldn't result in Rosy taking twice as much time. The other options are 'some' or 'tiny amount', which are not as accurate as 'moderate amount' in this context because they could mean a lesser extent of efficiency difference.

Therefore, the answer is **A. moderate amount**

[Qwen-1.5-7B-Chat] Rosy is 'x' times more efficient than Mary. So, the time taken by Rosy is $12 / (1 + x)$ days.

We are given that $12 / (1 + x) = 8$.

Let's solve for x: $(1 + x) = 12 / 8 = 1.5$

Simplifying x, $x = 0.5$.

This means that Rosy is half as efficient as Mary. So, 'some' is the closest quantifier choice for 'most'. If x were to be a tiny portion, this would not make sense. Therefore, the answer is **'some' (or B). #### B**

Table 7: Examples of Qwen-1.5-Chat models on FRoG where models of different parameters compute similar percentage values about the target percentage mention, but holding different opinions about the quantifier choice. The explicit quantifier estimation is highlighted.

Prompt Templates for FROG Problems

Given the (Question, Original Answer) of the original benchmark, the FROG problems can be constructed with the following templates.

Problem Template

Question:
{FROG Question}
Answer:
Let's think step by step

FROG Question Template

[MASK]:
{Question}
{Choices}
If the answer to the question is {Original Answer}, then please select the quantifier that is closest to the meaning of [MASK] from the following choices.

[Mislead]:
{Question}
{Choices}
If the answer to the question is {Original Answer}, and {Mislead Quantifier} in the question is not correct, please select the most appropriate quantifier from the following choices.

[X%]:
{Question}
{Choices}
If the answer to the question is {Original Answer}, then please select the most appropriate quantifier that is closest to the meaning of X% from the following choices.

[Mask Percentage]:
{Question}
{Choices}
If the answer to the question is {Original Answer}, then please select the percentage value that is closest to the meaning of [MASK] from the following choices.

Table 8: Prompt template for FROG problems.

Instruction used for FROG Evaluation

You are an expert in mathematical reasoning and generalized quantifier reasoning. Here you are asked to answer one mathematical question based on real-life scenarios with a description starting with 'Question:' For example, the question may describe the driving experience of a person. Your answer will start with 'Answer: let's think step by step'.

You will also be provided with four possible choices, please select the choice that is closest to your estimation of the answer.

The answer needs to include necessary reasoning steps to demonstrate your thinking procedure, and the final result of your calculation is demonstrated at the end of your answer starting with '####'.

Here are some examples starting with 'Question: ' for your reference.

Table 9: Instruction employed in FROG evaluation.