Numerical Sensitivity Enhancing and Reasoning Completeness Alignment for Quantitative Understanding

Xinyue Liang*, **Jiawei Li***, **Yizhe Yang, Yang Gao**†
School of Computer Science and Technology,
Beijing Institute of Technology, Beijing, China
Beijing Engineering Research Center of High Volume Language Information
Processing and Cloud Computing Applications, Beijing, China
Beijing Institute of Technology
Southeast Academy of Information Technology, Putian, Fujian, China
{xyliang, jwli, yizheyang, gyang}@bit.edu.cn

Abstract

In this paper, we describe the methods used for Quantitative Natural Language Inference (QNLI), and Quantitative Question Answering (QQA) in task1 of Semeval2024 NumEval. The challenge's focus is to enhance the model's quantitative understanding consequently improving its performance on certain tasks. We accomplish this task from two perspectives: (1) By integrating real-world numerical comparison data during the supervised fine-tuning (SFT) phase, we enhanced the model's numerical sensitivity. (2) We develop an innovative reward model scoring mechanism, leveraging reinforcement learning from human feedback (RLHF) techniques to improve the model's reasoning completeness. The experimental results demonstrate that our methods achieve outstanding performance. Our code could be found at https://github.com/Bit-numeval/NumEval.

1 Introduction

Numeral-aware language understanding plays a crucial role in narratives when contexts contain numerical comparison, implication, definition etc (Chen et al., 2023a). This importance is especially revealed in real-world applications, such as law, finance, and clinical documentation (Chen et al., 2024b). Large Language Models (LLMs) demonstrated impressive performance in various NLP tasks, but they still suffer from challenges such as hallucination of weak numerical reasoning (Zhuang et al., 2024). In this paper, we delve into solving the numerical hallucination problem, evaluated by the QNLI (Ravichander et al., 2019) and QQA (Mishra et al., 2022) subtasks in the Quantitative Understanding (QU) task (Ravichander et al., 2019) of the Semeval Numeval Task (Chen et al., 2024a). QNLI involves inferring relationships based on numerical clues, and QQA requires quantitative reasoning. Table 6 in Appendix A.1 shows examples of each task.

QQA

prompt: I will first raise a question and then provide two options. Please choose the correct answer after providing the inference process step by step, in the format of 'the answer is option 1. ####_1'.
question: "The ranger and the rustler both were riding horses that galloped at the same speed. The rustler left at 01:00 where as the ranger left at 05:00. Who has traveled further??"
option1: the ranger
option2: the rustler
Response: Let's think step by step.

label: the answer is option 2. #### 2

Solution with inaccurate steps
[1]. The ranger and the rustler both were riding horses that galloped at the same speed. The rustler left at 01:00 where as the ranger left at 05:00 hours. [2]. Since the ranger left at 05:00 hours and the rustler left at 01:00 hours, **the ranger has traveled for 4 hours more than the rustler.** [3]. As they were riding at the same speed, the ranger has traveled further in the given time period. [4]. Therefore, the answer is option 1. #### 1

Solution with incomplete reasoning
[1]. Given that both the ranger and the rustler were riding horses at the same speed. The rustler left at 01:00 and the ranger left at 05:00 hours. [2]. Therefore, the answer is option 2: the rustler. #### 2

Correct solution
[1]. Given that both the ranger and the rustler were riding horses at the same speed, the distance traveled is directly proportional to the time traveled. Distance = Speed * Time. [2] . The rustler left at 01:00 and the ranger left at 05:00 hours. Since the rustler started 4 hours earlier, the rustler has traveled for 4 hours more compared to the ranger. [3] . As they were riding at the same speed, the rustler has traveled further in the given time period. [4] . Therefore, the answer is option 2: the rustler.#### 2

Figure 1: An example from QQA dataset. LLMs may not be able to generate an accurate and complete process during quantitative reasoning. Specifically in this example, the first solution has an error in step[2] where the model confuses the concept of time period and time point, resulting in a wrong answer. And the second solution simply jump to the final answer after summarizing the problem, which is incomplete and unreasonable.

Based on our investigation and preliminary evidence of promise, we attribute LLMs' limitations on the QU tasks to two key aspects: (1) *Numerical Sensitivity*: LLMs, trained on vast quantities of text, often fail to accurately capture numerical information (Chen et al., 2023b). (2) *Reasoning Accuracy and Completeness*: as illustrated in Figure 1, LLMs may struggle to generate a comprehensive and precise step-by-step reasoning process, particularly in numerical reasoning contexts (Bílková et al., 2023).

To improve models' numerical sensitivity, Chen et al. (2023b) fine-tuned them using the Comparing

*Equal contribution.
†Corresponding author.

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 1830–1841

Numbers Dataset, which comprises numerical comparison statements. However, solely tuning models using the comparing number data may lead to an overfit issue. Meanwhile, recent efforts on enhancing reasoning accuracy such as process supervision by reinforcement learning on every reasoning step (Lightman et al., 2023). Nevertheless, in our cases, numerical reasoning involves a variable number of reasoning steps. Therefore, multiplying reward scores for each step (Lightman et al., 2023) reduces the overall multi-step reasoning score, which results in incomplete reasoning steps.

To address these limitations, we propose utilizing numerical comparisons of real-world contexts for more robust fine-tuning. In addition, we introduce a reasoning completeness reward designed to improve the precision of viable reasoning processes. The contributions of this paper include: (1) By integrating the comparing numbers task during the fine-tuning, we enhance the model's numerical sensitivity. Specifically, we use GPT-3.5 to integrate comparing numbers data into the real-world context, effectively preventing overfitting during training. Additionally, we reduce the long-tail effect by balancing between comparing numbers data and QU task data. Ablation studies show significant performance improvements with this method. (2) To the best of our knowledge, our study is the first time to enhance the model's reasoning completeness by RLHF. By introducing a fine-grained Reasoning Completeness Reward method, we emulate the complexity of human reasoning processes, aligning the model's accuracy and step rationality with human feedback. Experimental results confirm that our approach effectively improves the performance by ensuring a reasonable number of reasoning steps. (3) Our approach outperforms the other models of the same size across all test datasets, demonstrating strong generalizability. Furthermore, even compared to the state-of-the-art LLMs such as GPT-3.5 (Ouyang et al., 2022) and Llama2-70B (Touvron et al., 2023), our method also achieves better performance on four datasets.

2 System Overview

As shown in Figure 2, we highlight to enhance the model's *numerical sensitivity* and *reasoning completeness*. Specifically, we first use GPT-3.5 (Ouyang et al., 2022) to extend comparing numbers data into real-world contexts and fine-tune the model with this data to enhance numerical sensitivity. To prevent model overfitting, we mix 50% of the QNLI and QQA task data and 50% comparing numbers data into the SFT training dataset. Furthermore, we employ RLHF method to align every reasoning step with human-labeled process supervision. To leverage a more profitable Reward model for RLHF, we manually score the reasoning steps of the augmented positive and negative cases. In particular, we propose a newly Reasoning Completeness Reward for the PPO algorithm to encourage a complete reasoning procedure. The following subsections will detail our method.

2.1 Comparing Numbers Task for Numerical Sensitivity Enhancement

The comparing numbers task is proven to enhance the numerical sensitivity of the model (Chen et al., 2023b). Nevertheless, traditional comparing numbers data only involves the comparison of two numbers and lacks real-world contexts, which can lead to model overfitting and impairing its comprehension and generation capabilities. To address this, we use GPT-3.5 to put comparing numbers data into real-world contexts for training. Additionally, we introduce training data balance to avoid overfitting and long-tail problems. The following will provide a detailed explanation.

Comparing Numbers in Real-world Contexts The comparing numbers task was first proposed by Chen et al. (2023b), statements in the format "[Num 1] is equal to [Num 2], the answer is True/False." We further improve the statements by using GPT-3.5 to incorporate comparing numbers data into real-world contexts, thereby increasing the diversity and reality of the data. Additionally, randomly generating [Num 1] and [Num 2] overlooks the realistic numerical ranges in real-world contexts (e.g. human ages cannot reach 100,000 years old). Therefore, we restrict the numerical range to ensure that 90% of the numbers are randomly generated within the range of 0 to 10,000, thus aligning more closely with real-world contexts. Details and an example are shown in Appendix A.

Training Data Balance To balance the data and avoid long-tail problems caused by varying dataset sizes in QNLI and QQA tasks, we generate additional data by using GPT-3.5, which has increased the number of cases in each dataset to approximately 1000. Moreover, during the training phase of the comparing numbers task, we mix 50% of

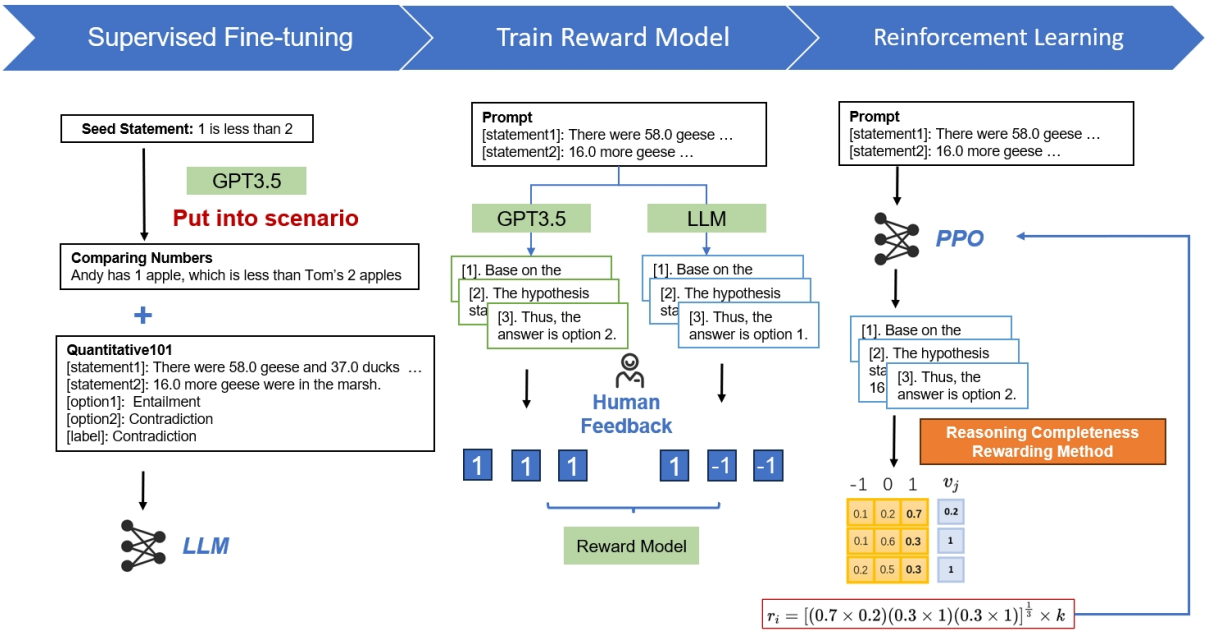

Figure 2: An overview of our system: (1) supervised fine-tuning with comparing numbers task for numerical sensitivity enhancement, (2) reward model training. (3) reinforcement learning via proximal policy optimization with Reasoning Completeness Reward.

the QU training data and 50% comparing numbers data to avoid overfitting the model to the comparing numbers task. Details on the specific expansion methods and prompt specifics can be found in the Appendix C.

2.2 RLHF-based Reasoning Completeness Alignment

To enhance the model's reasoning accuracy and completeness, we first employ human-labeled process supervision signals to align every reasoning step generated by the LLMs; then, we propose a new Reasoning Completeness Reward (RCR) model to improve the RLHF's performance to encourage generating complete reasoning steps.

2.2.1 Human-data Collection for Training Reward Model

To train a profitable reward model (RM), balanced labels need to be collected. While the number of positive labels far exceeds other labels among the steps generated by GPT-3.5, we have also used other open-source LLMs, such as Abel-7b, to generate candidates of reasoning steps, which may contain more negative examples to balance the labels' polarities. Human labelers would evaluate the given steps by their correctness, and correct answers to the question are provided as a reference. The statistics of datasets are shown in Table 1.

Datasets	Cases	Human labeled			
		Pos.	Neu.	Neg.	Steps
AwpNLI	1622	4334	822	1669	7109
NewsNLI	1643	3358	910	2870	7502
RedditNLI	1152	3074	507	958	4674
RTE_Quant	1324	3363	290	914	4817
StressTest	1369	2598	723	1921	5696
QQA	1394	3937	184	1778	6424

Table 1: The step data labeled by human. "Cases" is the number of solutions generated by models, "Pos.", "Neu.", and "Neg." are the number of positive, neutral, and negative labels after labeling, respectively, "Steps" is the total number of reasoning steps taken to solve all the questions in the dataset.

Step Labelling Criteria Each step is classified as either 'positive', 'neutral', or 'negative' due to its correctness, corresponding to three labels: "1", "0", and "-1". The correct steps must first meet the requirements of accurate logic and calculation within the steps (correct object of operation and correct result). At the same time, it is necessary to be consistent with and correctly use the results of the previous step for subsequent reasoning. If the correct conditions are met but there is no help in obtaining the correct answer, 0 points will be given. On this basis, if the task requirements are correctly understood and helpful in obtaining the correct answer, 1 point can be given. Steps with

logical, computational, or factual errors that are completely unrelated to the context and question, or incorrect answers, will receive a score of -1.

2.2.2 Reasoning Completeness Reward

Lightman et al. (2023) proposed a process supervision method by scoring the correctness probability of each reasoning step. The score is implemented as the multiplication of probabilities of all reasoning steps:

$$r_i^1 = \prod_{j=1}^{N} P(y_j = 1|x_j) \tag{1}$$

where N is the number of steps for the i-th solution, x_j is the input of RM, and y_j is the classification.

However, when the number of reasoning steps is not fixed, the score of (1) is influenced by the number of reasoning steps. As the correctness probability is decimal, the more steps involved in reasoning, the smaller the product of probabilities, resulting in lower rewards, which leads to a tendency for the model to subsequently generate less reasoning steps. To mitigate this, we applied geometric mean to the product:

$$r_i^2 = \left(\prod_{j=1}^{N} S_j \right)^{\frac{1}{N}} \tag{2}$$

where $S_j = P(y_j = 1|x_j)$ is the score for j-th step.

We observed that despite using the scoring method of (2), the model still failed to generate complete reasoning steps. Further analysis revealed that the model often simply repeats the question in its first reasoning step, resulting in a high score for the first step, which in turn leads the model to refrain from generating subsequent steps. Therefore, we propose the **reasoning completeness reward**, including a weighted geometric mean and a penalty coefficient. First, the importance of steps at different positions can be adjusted by setting weight v_j.

$$r_i^3 = \left(\prod_{j=1}^{N} v_j S_j \right)^{\frac{1}{N}} \tag{3}$$

In addition, as we hope that the solutions are around 4 steps, and solutions guessing the result from the first step without reasoning is not encouraged, a penalty coefficient k is introduced to constrain it.

$$k = \begin{cases} \frac{5}{\sigma\sqrt{2\pi}} e^{-\frac{(N-\mu)^2}{2\sigma^2}} & , N > 1 \\ 0 & , N \leq 1 \end{cases} \tag{4}$$

where $\mu = 4$, $\sigma = 2$. So the reward from the reward model is

$$R_i = r_i - \beta KL(x, y) \tag{5}$$

$$r_i^4 = \left(\prod_{j=1}^{N} v_j S_j \right)^{\frac{1}{N}} \times k \tag{6}$$

where $KL(x, y)$ is the KL-divergence between the current policy and the reference model in reinforcement learning.

Upon achieving the RM, we employ RLHF with PPO (Schulman et al., 2017) in a step-by-step manner, which is implemented with TRL[1].

3 Experimental Setups

Datasets We adopted the Quantitative101 dataset provided for SemEval 2024 Task7 and then expanded it using the GPT-3.5 API, in Table 1. From these data, three datasets were obtained for SFT, reward model training, and reinforcement learning, respectively. The prompts used during training and testing can be found in the Appendix D. Due to a large amount of labeled "1" data in the RM training dataset, each step of "0" and "-1" was repeated 2-3 times, resulting in 16587 positive steps, 11072 neutral steps, and 16236 negative steps in total. When dividing the datasets, 20% of the data is used as test sets in all three periods.

Metrics and Parameters setting The metric is the average micro-F1 score of the testing dataset in QNLI and QQA tasks. Our CN-SFT model is trained on Abel-7B (Chern et al., 2023) with a learning rate of 3e-5, a warmup rate of 0.03, and a model max length of 1024. As for the RM, we choose to train on BERT-large model (Devlin et al., 2018) as it well complete the classification tasks (Gao et al., 2022). It is trained with a learning rate of 2e-5, warmup rate of 0.05, and a model max length of 256, and is trained for 10 epochs. The PPO training is implemented with Lora, where the learning rate=1.41e-5, max new tokens=512. On a dataset of size 5470, each training epoch takes around 55 hours on 4 A100s.

4 Experimental Results

4.1 Overall Results

Main Results Table 2 compares the performance of our method with that of current mainstream

[1] https://huggingface.co/docs/trl/main/en/index

Models	QNLI					QA	Score
	AwpNLI	NewsNLI	RedditNLI	RTE_Quant	StressTest		
Llama-7B	1.47%	0.47%	0.40%	0.86%	1.36%	3.70%	1.38
GPT-3.5	42.07%	58.55%	32.0%	55.88%	33.1 %	40.12%	43.62
BLOOMZ	48.04%	54.46%	37.2%	47.64%	31.22%	51.85%	45.07
Abel-7B	55.82%	50.75%	47.20%	56.67%	30.87%	48.14%	48.24
ChatGLM	72.55%	70.42%	55.2%	60.94%	37.15%	53.70%	58.33
GPT-3.5*	77.93%	51.3%	59.2%	73.53%	**54.77%**	**63.58%**	63.39
Llama-70B	77.45%	69.01%	67.2%	73.39%	37.15%	59.26%	63.91
CN-SFT-7B	71.08%	66.67%	64.40%	72.53%	52.74%	56.17%	63.93
CN-PPO-7B	**87.25%**	**71.36%**	**75.20%**	**86.99%**	53.57%	56.68%	**71.84**

Table 2: Performance of baseline models. The prompt of GPT-3.5* has added explanations for options such as "entailment" compared to GPT-3.5. The CN means comparing numbers. CN-PPO-7B is trained on CN-SFT-7B with RCR-improved RLHF.

Dataset	Lightman et al. (2023)	Ours
AwpNLI	83.33.%	**87.25%**
NewsNLI	69.95%	**71.36%**
RedditNLI	63.20%	**75.20%**
RTE_Quant	**88.41%**	86.99%
StressTest	37.32%	**53.57%**
QQA	51.23%	**56.68%**
Score	65.57	**71.84**
Steps (avg)	2.624	**2.844**

Table 3: Comparison results indicate that our proposed RCR-improved RLHF outperforms over all datasets and can generate more completed reasoning steps.

LLMs on the QU tasks. Our model achieved optimal performance in the AwpNLI, NewsNLI, RedditNLI, and RTE_Quant tasks. It also showed comparable results in the StressTest and QA tasks, only falling short of Llama2-70B and GPT-3.5. However, it is worth emphasizing that our model has only 7B parameters. At this scale, its performance significantly surpasses that of other models.

Specifically, compared to our baseline model Abel-7B, by solely employing the CN-SFT method, our model achieved significant accuracy improvements of 15.26%, 15.92%, 17.12%, 15.86%, 21.87%, and 8.03% across six tasks. Upon further integrating the RLHF, the accuracy additionally gained 16.17%, 4.96%, 10.8%, 14.46%, 0.83%, and 0.51% improvement. These results validate the effectiveness of the methods proposed in this study.

The Effect of the Reasoning Completeness Reward (RCR) It is aimed at enhancing the completeness of the reasoning steps. Table 3 shows the comparison of our method's effectiveness and the number of reasoning steps against the baseline. The results demonstrate that the proposed RCR significantly increases the performance. Furthermore, the number of reasoning steps generated by our proposed enhances the reasoning completeness indicated by reasoning steps.

4.2 Ablation Analysis

We further conduct ablations to analyze the contribution of our methods' components.

Comparing numbers task can enhance the model's numerical sensitivity. We first verify whether the comparing numbers task enhances the model's numerical sensitivity. As shown in Table 4, By comparing the results of SFT (column 2) and CN-SFT (column 3) as well as PPO (column 4) and CN-PPO (column 5), we observe that models integrating the comparing numbers task exhibit superior performance in all datasets.

RLHF-based reasoning completeness alignment is valid. As shown in Table 4, the comparison of the PPO (column 4) to the SFT (column 2), and the comparison of the CN-PPO (column 5) to the CN-SFT (column 3) indicate that reasoning completeness alignment based on the proposed RLHF can effectively improve the model's performance on numerical understanding.

4.3 Comprehensive Analysis

4.3.1 Error Analysis

As shown in Table 2, the system performs relatively weakly on the QQA and StressTest datasets. The weak accuracy in the QQA task may be attributed to a lack of physical common sense in our 7B LLM. For instance, the question `"An apple is`

Dataset	SFT (w/o CN and RL)	CN-SFT (w/o RL)	PPO (w/o CN)	CN-PPO
AwpNLI	58.82%	**71.08%**	80.67%	**87.25%**
NewsNLI	55.87%	**66.67%**	59.62%	**71.36%**
RedditNLI	51.60%	**64.40%**	71.20%	**75.20%**
RTE_Quant	68.40%	**72.53%**	80.72%	**86.99%**
StressTest	52.57%	**52.74%**	53.40%	**53.57%**
QQA	50.62%	**56.17%**	**59.26%**	56.68%
Score	56.31	**63.93**	67.48	**71.84**

Table 4: Ablation studies of our method. SFT means the model is fine-tuned only on QU training data, while PPO refers to reinforcement learning training based on this model. CN-SFT means the model was fine-tuned on both QU training data and comparing numbers data, and CN-PPO refers to reinforcement learning training based on this model.

`sitting 15 meters away from Harry, and a watermelon is sitting 110 cm away. Which item looks larger?`". Another example is shown in Appendix B.1. Solving such a problem not only relies on numerical logical reasoning, but also requires understanding the conversion relationship between 'meters' and 'cm', and the physical principle that objects appear smaller the further away they are. This common sense is often acquired by knowledge injection for LLMs, which is out of our research scope in this paper.

The objective of the StreesTest dataset is to determine the relations of two sentences. Most of the sentences always contain multiple numbers whereas only one or two numerical information is valuable for classifying the sentences' relations. An example is shown in Appendix B.2. However, our models as well as other LLMs (i.e. GPT3.5 and Llama-70B) hardly capture the most valid numbers to predict the outcomes. As a result, the improvement of our model on the StressTest dataset is not as significant as in other datasets.

4.3.2 Strengths and Weaknesses

Strengths This study firstly integrates comparing numbers data into real-world contexts, thereby avoiding model overfitting and the deterioration of linguistic capabilities typically caused by solely using formatted data. This approach not only enhances the model's numerical sensitivity but also effectively prevents overfitting issues. Moreover, we propose a new reasoning completeness reward scoring method, suitable for more complex reasoning tasks, particularly those featuring a variable number of reasoning steps. The effectiveness of this method lies in rewarding each step of reasoning and considering the number of reasoning steps into the reward calculation, thus preventing the generation of reasoning processes that are either too brief or excessively lengthy. Finally, In the majority of tasks, our 7B model outperforms super LLMs such as GPT-3.5 (Ouyang et al., 2022) and Llama2-70B (Touvron et al., 2023).

Weaknesses First, in Section 4.3.1, We noted that the model generates incorrect answers for certain tasks due to the absence of essential physical common sense and demonstrates suboptimal performance in identifying and predicting relationships involving multiple numbers. Second, our approach substantially mitigates the model's hallucination of weak numerical reasoning but doesn't eliminate the hallucination that existed in LLMs' outcomes. Third, this study employs the PPO algorithm for the RLHF to validate its effectiveness. Nevertheless, the learning efficiency and convergence problems of the PPO algorithm have not been fully explored.

Therefore, future work is directed to the following aspects. The first one is knowledge injection (Lauscher et al., 2020; von Rueden et al., 2023), especially numerical-relevant knowledge, could be further employed to improve the numerical-aware language understanding capability of the LLMs. Second, the most valuable numbers during the reasoning process could be identified and weighted. Third, employing Score Normalization and Clipping to constrain the reward scores can resolve the training instability (Zheng et al., 2023). Last, utilizing the DPO algorithm (Rafailov et al., 2023), which implements an implicate reward, enhances training stability and its efficiency.

5 Conclusion

In this paper, we described the systems used for QNLI, and QQA in task1 of Semeval2024 NumEval. We select the Abel-7B model as the baseline model. To address the quantitative understanding problem, we first integrate comparing numbers data from real-world contexts to enhance the model's numerical sensitivity. During this process, we devise an effective data mixer to prevent overfitting and the long-tail problem. Subsequently, by employing process supervision from human feedback, we develop an innovative reward model scoring mechanism to improve the model's reasoning completeness using RLHF. Test results demonstrate that our 7B model exceptionally outperformed, surpassing LLMs such as GPT-3.5 on 4 tasks and Llama2-70B on 6 tasks, respectively.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (Grant No: 92370110, U21B2009). We appreciate the helpful discussions with Siming Liu. We also thank all the anonymous reviewers for their insightful suggestions.

References

Marta Bílková, Sabine Frittella, Daniil Kozhemiachenko, and Ondrej Majer. 2023. Qualitative reasoning in a two-layered framework. *International Journal of Approximate Reasoning*, 154:84–108.

Chung-Chi Chen, Jian-Tao Huang, Hen-Hsen Huang, Hiroya Takamura, and Hsin-Hsi Chen. 2024a. Semeval-2024 task 7: Numeral-aware language understanding and generation. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*. Association for Computational Linguistics.

Chung-Chi Chen, Yu-Lieh Huang, and Fang Yang. 2024b. Semantics matter: An empirical study on economic policy uncertainty index. *International Review of Economics Finance*, 89:1286–1302.

Chung-Chi Chen, Hiroya Takamura, Ichiro Kobayashi, and Yusuke Miyao. 2023a. Improving numeracy by input reframing and quantitative pre-finetuning task. In *Findings of the Association for Computational Linguistics: EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 69–77. Association for Computational Linguistics.

Chung-Chi Chen, Hiroya Takamura, Ichiro Kobayashi, and Yusuke Miyao. 2023b. Improving numeracy by input reframing and quantitative pre-finetuning task. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 69–77, Dubrovnik, Croatia. Association for Computational Linguistics.

Ethan Chern, Haoyang Zou, Xuefeng Li, Jiewen Hu, Kehua Feng, Junlong Li, and Pengfei Liu. 2023. Generative ai for math: Abel. `https://github.com/GAIR-NLP/abel`.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Leo Gao, John Schulman, and Jacob Hilton. 2022. Scaling laws for reward model overoptimization.

Anne Lauscher, Olga Majewska, Leonardo FR Ribeiro, Iryna Gurevych, Nikolai Rozanov, and Goran Glavaš. 2020. Common sense or world knowledge? investigating adapter-based knowledge injection into pretrained transformers. *arXiv preprint arXiv:2005.11787*.

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

Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan. 2022. NumGLUE: A suite of fundamental yet challenging mathematical reasoning tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3505–3523, Dublin, Ireland. Association for Computational Linguistics.

Long Ouyang, Jeffrey 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 F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model.

Abhilasha Ravichander, Aakanksha Naik, Carolyn Rose, and Eduard Hovy. 2019. EQUATE: A benchmark evaluation framework for quantitative reasoning in natural language inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 349–361, Hong Kong, China. Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms.

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

Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Laura von Rueden, Jochen Garcke, and Christian Bauckhage. 2023. How does knowledge injection help in informed machine learning? In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Rui Zheng, Shihan Dou, Songyang Gao, Yuan Hua, Wei Shen, Binghai Wang, Yan Liu, Senjie Jin, Qin Liu, Yuhao Zhou, Limao Xiong, Lu Chen, Zhiheng Xi, Nuo Xu, Wenbin Lai, Minghao Zhu, Cheng Chang, Zhangyue Yin, Rongxiang Weng, Wensen Cheng, Haoran Huang, Tianxiang Sun, Hang Yan, Tao Gui, Qi Zhang, Xipeng Qiu, and Xuanjing Huang. 2023. Secrets of RLHF in large language models part I: PPO. *CoRR*, abs/2307.04964.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. 2024. Toolqa: A dataset for llm question answering with external tools. *Advances in Neural Information Processing Systems*, 36.

A Construction Process for Our Comparing Numbers Task

To create our Comparing Numbers data, we first automatically generate seed statements and then put them into natural language paragraphs by GPT3.5.

As Table 5 shows, there are three templates for seed statements. We randomly select two numbers from 0 to 9,999 and insert them into the template, note that the distributions of each template and answers are balanced. Finally, 5059 instances are obtained, small amount of duplication in numbers is acceptable as they will be placed into different scenarios afterwards.

Considering most scenarios in the QU tasks are daily situation and financial news, we adopted the following two prompts to generate statements respectively.

Prompt for daily situations: Rewrite the sentence containing numerical comparison relationships into a paragraph describing daily situations about numbers, with a length of no more than 50 words, comparative relationships must be included: (seed statement). For example : 'There were 128,695 students in the large university, which exceeded the 107,736 count of another university.'

Prompt for financial news: Rewrite the sentence containing numerical comparison relationships into a paragraph of financial news, with a length of no more than 50 words, comparative relationships must be included : (seed statement) For example: 'In the stock market, stock A's price at 183.146 increase, surpassing stock B's price at 115.877.'

A.1 Examples of Different Tasks

As shown in Table 6, the comparing numbers task involves a statement with a numerical relationship, which requires the model to determine if it is true.

In the QNLI task, there are two statements, the first is the premise, and the second is the hypothesis. The model needs to determine the correct relationship (entailment/neutral/contradiction) between the two statements, that is, to determine whether the hypothesis can be inferred from the premise. In the QQA task, there is a question with two options, and the model's task is to work out the correct answer.

These tasks require models to interpret quantities expressed in language, perform basic calculations, judge their accuracy, and justify quantitative claims using both verbal and numeric reasoning.

B Examples of Model Results

B.1 Example from QQA task

As shown in Table 7, in QQA tasks, the model sometimes becomes confused about the knowledge required for this problem, unable to analyze based on common sense that lightweight paper airplanes can fly faster, but instead conducts analysis unrelated to the problem, resulting in incorrect answers.

B.2 Example from StressTest

From Table 8 we can see that although the model correctly extracted quantitative information, it misses the key numeral and is distracted by the text, conducting calculations unrelated to the question, resulting in wrong answer.

C Dataset Extending

C.1 QNLI tasks

For the QNLI task, first automatically generate a set of numerals which will be contained by the premise and generate the premise with GPT3.5, then rewrite the statement based on the "entailment", "neutral"or "contradiction"relationship as hypotheses.

For example, when expanding the NewsNLI dataset, we use the following prompts in sequence.

To generate a premise: " Write a piece of news in 30 words or less that contains the message "[number]"

To generate an entailed hypothesis: "Abbreviate this paragraph and keep its original meaning unchanged:" [premise] "

To generate a neutral hypothesis: " Add some numerical information to this paragraph:[statement] "

If a contradicted statement needs to be generated, simply replace the numbers in the premise, such as replacing" 30 people "with" 40 people "," more than 50 people ", or" less than 20 people ".

When expanding the AwpNLI dataset, we can first generate a pair of statements with entailment relationships, the prompt is as follows: "Generate two statements, the first being a promise that contains some quantitative information, and the second statement is a quantitative inference based on the premise. For example: [statement1]: A restaurant baked 5.0 cakes during lunch and sold 6.0 during dinner today and the restaurant baked 3.0 cakes yesterday." [statement2]: 2.0 cakes are left," and

Seed statement	Question	Label
200 **is less than** 215	At the cafe, a line of 215 individuals formed, exceeding the queue at the bakery, where 200 people were waiting.	True
83.146 **is larger than** 115,899	In the stock market, stock A's price at 83.146 increase, surpassing stock B's price at 115,877.	False
147,254 **is equal to** 32.567	There were 147,254 votes for candidate A, which was equal to 32,567 votes for candidate B.	False

Table 5: Examples of our Comparing Numbers task. The seed statements and labels are generated by randomly selecting two numbers between 0 and 9,999 to create comparison statements. The questions are then formulated by GPT-3.5 by specific prompts.

Task	Question	Label
Comparing Numbers	At the cafe, a line of 200 individuals formed, exceeding the queue at the bakery, where 215 people were waiting.	True/False
QNLI	statement1: The fighting ended with all seven attackers dead, Afghan officials said. statement2: All seven militants are dead , authorities say.	Entailment/ Contradiction/ Neutral
QQA	The beauty queen glided across the marble floors with a speed of 15 m/s but at a speed of 1000 cm/s on the wooden floors because it had? Option1: more resistance. Option2: less resistance	Option1

Table 6: Examples of Comparing Numbers, QNLI, and QQA task.

then replace the numerical information in statement 2 to obtain the conflicting statement.

C.2 QQA task

For QQA tasks, most of the questions in the dataset are simple physics multiple choice questions, including knowledge of friction and kinematics. To maintain a balanced proportion of answers, specify the answer option in the prompt of the generated question: "Generate a physics multiple choice question with a quantitative relationship, consisting of two options. The correct answer is option [ans], avoid using markers or latex markers. For example:" question ":" James is watching it rain one day. He notes that the rain travels over the driveway in 5 seconds and the bump "shingles of his house's roof in 10 seconds, the driveway and the shingles are all similar in size. Which surface is more smooth?," option1 ":" roof "," option2 ":" driveway. ".

D Prompts Used in Our Work

Table 9 shows all the prompts we used during the training process.

QQA
question: Two paper planes are thrown at a time, into the air. One is made of thick cardboard and has a weight of 200 gms, one is made of thin paper and weighs 40 gms. Which paper plane flies faster? **option1:** cardboard plane **option2:** paper plane **label:** Thus, the answer is option2. #### 2
SFT output [1]. The flight speed of a paper plane could be given by the equation: S = faultincline * cos(theta) * times filled, where fill time = t, S is the flight speed, and proprietor elevation. [2] . As the Fuji Schoolcogel 2424 AppColl Maybe is a smaller number, we can assume it's an inertial force. [3] . We can calculate the weight using the equation: W = m * g / W, where W is the wheel weight, m is the mass, g is the acceleration, and D is the diameter of the wheel. [4] . Finally, we can determine the acceleration of a cardboard plane using the equation: Continuous acceleration = (m * g) / times filled, where W is the weight, m is the mass, and D is the diameter of the wheel, w_c = 200 g. [5]. Calculating the acceleration of the cardboard plane: Continuous acceleration = (200 g * g) / 10 sec = 200 g * g / 10 sec = 200(g) / 10 sec. [6] . Calculating the acceleration of the thin paper plane: Continuous acceleration = (40 g * g) / 10 sec = 40 g * g / 10 sec = 40(g) / 10 sec. [7] . There is no difference in acceleration between the cardboard plane and the thin paper plane, so they will fly at the same speed. Thus, the answer is option 1. #### 1

Table 7: An error example from QQA task. Model sometimes becomes confused about the knowledge required for this problem, unable to analyze based on common sense that lightweight paper airplanes can fly faster, but instead conducts analysis unrelated to the problem, resulting in incorrect answers.

StressTest
statement1: If Danny **scored 86** , 75 , 52 , 87 , 85 and 82 marks (out of 100) in English , Social Studies , Art , Music , Biology and French **statement2:** If Danny **scored more than 76** , 75 , 52 , 87 , 85 and 82 marks (out of 100) in English , Social Studies , Art , Music , Biology and French **option1:** Entailment **option2:** contradiction **option3:** neutral **label:** Thus, the answer is option1. #### 1
CND-PPO output [1]: Given the numbers 86, 75, 52, 87, 85, and 82, it can be calculated that the average mark is (86 + 75 + 52 + 87 + 85 + 82) / 6 = 485 / 6 = 77.5.[2] : The hypothesis states that Danny scored more than 76, 75, 52, 87, 85, and 82 marks.[3] : As the average mark approximately equals 77.5, and each mark is rounded to the nearest whole number before factoring, the hypothesis is 'not car more than 76, 75, 52, 87, 85, and 82 marks' in the given context.Thus, the answer is option 3. #### 3 **reward:** 0.28299634026503834
Correct solution [1]: The premise states that Danny scored 86 , 75 , 52 , 87 , 85 and 82 marks (out of 100). The hypothesis states that Danny scored more than 76, 75, 52, 87, 85, and 82 marks. [2]: **86 is indeed more than 76**, so the hypothesis can be infered to be true. [3] Thus, the answer is option 1. #### 1

Table 8: An error example from StressTest dataset. Although the model correctly extracted all the quantitative information, it misses the key point and conducted analysis and calculations unrelated to the question, resulted in wrong answer.

QQA I will first raise a question and then provide two options. Please choose the correct answer after providing the inference process step by step, in the format of 'the answer is option 1. #### 1'. If calculation is involved, please provide the equations during the calculation process. Using numbers like '1.' or '[1]' to mark steps. question: option1: option2: Response: Let's think step by step.
AwpNLI I will first raise two statements and then provide two options which are entailment and contradiction. The first statement is the given premise, the second statement is the hypothesis. You should determine if the hypothesis can be justifiably inferred to be true (option 1 : entailment) or false (option 2 : contradiction) base on the premise. Please choose the correct answer after providing the inference process step by step, in the format of 'the answer is option 1. #### 1'. If calculation is involved, please provide the equations during the calculation process. Using numbers like '1.' or '[1]' to mark steps. Choose the correct answer in the format of 'the answer is option 1. #### 1'. statement1: statement2: option1: option2: Response: Let's think step by step.
NewsNLI I will first raise two statements and then provide two options which are entailment and neutral. The first statement is the given premise, the second statement is the hypothesis. You should determine if the hypothesis can be justifiably inferred to be true (option 1: entailment) or cannot be determined (option 2: neutral) base on the premise. You should pay attention to additional information rather than shared information, especially paying attention to whether the numbers are reasonable and derived from the premise. If there is information that is not mentioned in the premise or cannot be directly inferred from the hypothesis, then the hypothesis cannot be determined. Please choose the correct answer after providing the inference process step by step, in the format of 'the answer is option 1 #### 1'. Using numbers like '1.' or '[1]' to mark steps. statement1: statement2: option1: option2: Response: Let's think step by step.
RTE I will first raise two statements and then provide two options which are entailment and neutral. The first statement is the given premise, the second statement is the hypothesis. You should determine if the hypothesis can be justifiably inferred to be true (option 1 : entailment) or cannot be determined (option 2 : neutral) base on the premise. You should pay attention to additional information rather than shared information, especially paying attention to whether the numbers are reasonable and derived from the premise. If there is information that is not mentioned in the premise or cannot be directly inferred in the hypothesis, then the hypothesis cannot be determined. Please choose the correct answer after providing the inference process step by step, in the format of 'the answer is option 1. #### 1'. Using numbers like '1.' or '[1]' to mark steps. statement1: statement2: option1: option2: Response: Let's think step by step.
RedditNLI I will first raise two statements and then provide three options which are entailment, contradiction and neutral. The first statement is the given premise, the second statement is the hypothesis. You should determine if the hypothesis can be justifiably inferred to be true (option 1 : entailment), false (option 2 : contradiction) or cannot be determined (option 3 : neutral) base on the premise. Please choose the correct answer after providing the inference process step by step, in the format of 'the answer is option 1. #### 1'. Using numbers like '1.' or '[1]' to mark steps. statement1: statement2: option1: option2: option3: Response: Let's think step by step.
StressTest I will first raise two statements and then provide three options which are entailment, contradiction and neutral. The first statement is the given premise, the second statement is the hypothesis. You should determine if the hypothesis can be justifiably inferred to be true (option 1 : entailment), false (option 2 : contradiction) or cannot be determined (option 3 : neutral) base on the premise. You should especially pay attention to whether the numbers are reasonable and derived from the premise. If there is information that is cannot be directly inferred in the hypothesis, then the hypothesis cannot be determined. Please choose the correct answer after providing the inference process step by step, in the format of 'the answer is option 1. #### 1'. Using numbers like '1.' or '[1]' to mark steps. statement1: statement2: option1: option2: option3: Response: Let's think step by step.

Table 9: Our prompts used for different datasets in the training process.