CTYUN-AI at SemEval-2024 Task 7: Boosting Numerical Understanding with Limited Data Through Effective Data Alignment

Yuming Fan* and **Dongming Yang** * † and **Xu He**
fanym@chinatelecom.cn , yangdongming@pku.edu.cn,
hex30@chinatelecom.cn,
China Telecom
Cloud Technology Co., Ltd

Abstract

Large language models (LLMs) have demonstrated remarkable capabilities in pushing the boundaries of natural language understanding. Nevertheless, the majority of existing open-source LLMs still fall short of meeting satisfactory standards when it comes to addressing numerical problems, especially as the enhancement of their numerical capabilities heavily relies on extensive data. To bridge the gap, we aim to improve the numerical understanding of LLMs by means of efficient data alignment, utilizing only a limited amount of necessary data. Specifically, we first use a data discovery strategy to obtain the most effective portion of numerical data from large datasets. Then, self-augment is performed to maximize the potential of the training samples. Thirdly, answers of all training samples are aligned based on some simple rules. Finally, our method achieves the first place in the competition, offering new insights and methodologies for numerical understanding research in LLMs.

1 Introduction

In recent years, the field of Natural Language Processing (NLP) has witnessed remarkable advancements, particularly with the advent of generative large language models (Jiang et al., 2023; Bai et al., 2023; Yang et al., 2023; Brown et al., 2020). These models have predominantly focused on textual data, demonstrating impressive capabilities in understanding and generating human-like text. However, an often-overlooked aspect of these developments is the nuanced role that numerical data plays in fully grasping the semantics of language. This oversight becomes particularly glaring in specialized fields such as stock market analysis, medical diagnostics, and legal decisions (Cortis et al., 2017; Modi et al., 2023; Jullien et al., 2023). In these domains, subtle numerical differences can have far-reaching implications, significantly affecting outcomes and decisions. Thus, the ability to understand and work with numbers, in these contexts underscores a critical gap in the semantic understanding capabilities of current language models.

Acknowledging this deficiency, there has been a growing interest within the NLP community towards enhancing the textual numeracy and computational abilities (Huang et al., 2023) of language models. This burgeoning interest has culminated in the introduction of SemEval2024's Shared Task 7. This innovative task is strategically designed to elevate the standards in the field by promoting the development of models that excel not only in literacy but also in computational skills. Such models promise to significantly boost usefulness and efficiency across a wide array of applications, ranging from automated financial analysis to predictive healthcare diagnostics and beyond.

However, the enhancement of numerical capabilities of LLMs heavily relies on the inclusion of a large amount of data, posing two significant challenges. On one hand, obtaining high-quality numerical annotated data is costly, as it requires significant economic costs and manual effort from professional annotators. On the other hand, the extensive use of as much data as possible to train the model can diminish the utility of high-value data and lead to increased computational consumption.

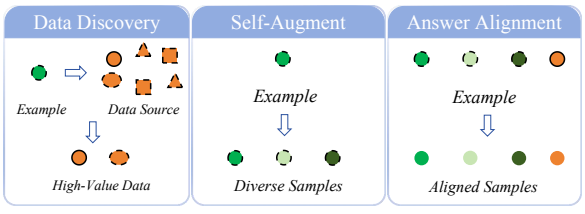

Figure 1: Pipeline of our system in this task.

As illustrated in Figure 1, this paper focuses on how to efficiently use a limited amount of data to

*Equal Contribution.
†Corresponding Author.

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 47–52

improve the numerical capabilities of LLMs. Our goal is to enhance the model's performance in numeral while minimizing costs, such as data annotation expenses and computational requirements for model training. Our method propose effective data alignment by employing strategies of data discovery, self-augment and answer alignment. The contributions are summarized as follows:

- A strategy of data discovery is proposed to extract numerical training samples, obtaining the most effective portion of numerical data from datasets and minimizing training costs.
- We implement original self-augment to all the training samples to maximize their effectiveness in enhancing the numerical capabilities of LLM.
- We align answers of all training samples according some customized rules to improve LLM's numerical reasoning performance and shorten the reasoning path.

After conducting numerous experiments and iterating on our strategies, we are proud to announce that we have secured the championship title at the competition of SemEval-2024 Task 7. Detailed ablation study and analysis of our method are also provided in this paper to identify contributions from individual components and facilitate future research.

2 Background

GPT-3 (Brown et al., 2020) marked significant progress in large language models, enhancing few-shot learning and demonstrating robustness across diverse NLP datasets. Bai et al. (Bai et al., 2023) developed the qwen model, notable for its performance in various tasks, particularly its chat model refined through human feedback. However, these models largely focus on textual data, paying limited attention to the importance of numerical values in semantic understanding.

Addressing this, the NumHG (Huang et al., 2023) dataset was introduced, focusing on generating news headlines with numerical information. Evaluations of high-performing models indicated room for improvement in numerical accuracy, aiming to advance research in numerically-focused headline generation and improve task performance.

Additionally, learning Mathematical Reasoning for tasks like GSM8K (Cobbe et al., 2021) and MetaMATH (Yu et al., 2023) remains a significant challenge for LLMs. Enhancing LLM reasoning through augmented output sequences (Wei et al., 2022) has been explored, with methods like Complexity-based CoT (Fu et al., 2022) showing that increased in-context steps can improve performance. Self-Consistency approaches (Wang et al., 2022) use multiple reasoning paths and majority voting to select answers. Other works leverage closed-source LLMs (Brown et al., 2020) for knowledge distillation (Magister et al., 2022), while some apply rejection sampling for better reasoning (Yuan et al., 2023). Techniques like the reinforced evol-instruct method (Luo et al., 2023) and constraint alignment loss for calibration (Wang et al., 2023) also contribute to the advancement of LLMs in mathematical reasoning.

Building on these developments, our work introduces a novel approach to refine LLMs' numerical reasoning capability. We fine-tune our base model with a curated selection of numerically samples, focusing on diversity and efficiency to cover a broader range of mathematical concepts.

3 System Overview

In this section, we will introduce our proposed method from several aspects. We start with data analysis of SemEval-2024 Task 7. Then, we present the proposed data discovery, self-augment and alignment strategies.

3.1 Data Analysis

The competition dataset, NumHG, provided news articles with headlines, where the task involved identifying masked numerical values in the headlines and explaining the calculations behind these numbers. Each data sample from NumHG comprises four elements: News, masked headline, calculation, and answer. As shown in Table 1, we conducted an analysis of the mathematical processing utilized in each data sample, and discovered the following: (1) Most answers can be directly copied from the text, indicating that these numerical values are explicitly mentioned. (2) Additionally, a portion of the answers required converting textual descriptions into numerical forms, involving text understanding and translation. (3) Simple mathematical operations, such as basic arithmetic and rounding, are also involved in a small subset of the dataset, demanding LLM to perform context-based mathematical operations.

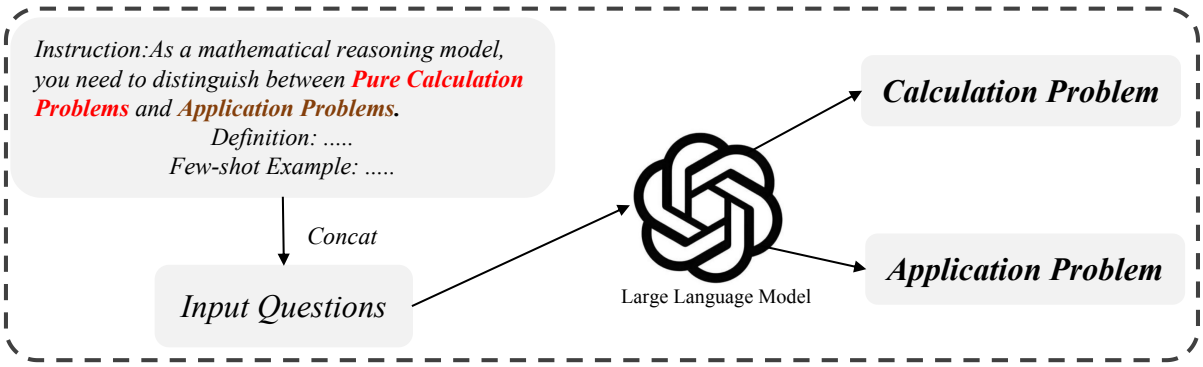

Figure 2: Data Discovery: Demonstrates the selection and integration of applicable problems from GSM8K and MetaMathQA datasets into the training set.

Example 3.1 Application Problem

At Rosa's Rose Shop, a bouquet containing a dozen roses costs $\$20$. If the price of a bouquet is directly proportional to the number of roses it contains, A bouquet of ____ roses will cost 65 dollars. What is the value of unknown variable ____?

Example 3.2 Pure Calculation Problem

Factor completely: $x^6 - 3x^4 + 3x^2 - ____$. The answer is 3. What is the value of unknown variable ____?

What is the value of ____ \times (7 - 5) - 5$? The answer is 1. What is the value of unknown variable ____?

Figure 3: Example of mathematical application problems and pure calculation problems in our dataset.

Table 1: Analysis on mathematical processing in the NumHG dataset.

Mathematical Processing Type	Count
Copy	15998
Trans	4111
Paraphrase	1727
Round	716
Subtract	496
Add	408
Span	104
Multiply	81
Divide	51
SRound	37

The above analysis reveals that the dataset and task possess distinct characteristics (e.g., emphasize understanding numbers within text rather than solving complex mathematical problems), suggesting that limited relevant data could potentially aid in enhancing performance. Furthermore, the high similarity among samples in this dataset underscores the importance of effective data augmentation and alignment strategies to maximize the utility of training samples.

To undertake the aforementioned investigation, we employed Qwen-Chat (Bai et al., 2023) as our base model, setting the input as a concatenation of news and masked headline, and output as a combination of calculation and answer to compile our training set. We crafted a preset prompt, *You are a numerical reasoning model. Please compute the correct number to fill the blank in a news headline.*, to utilize the inherent command-following ability of the LLM.

3.2 Data Discovery

As mentioned previously, we advocate for extracting a limited yet most effective subset of the dataset to enhance model performance. Specifically, we utilized the GSM8K and MetaMathQA datasets as the complementary source of external training data.

It is noted that, deviating from standard math tasks, the NumHG dataset focuses on understanding numerical semantics rather than complex calculations, primarily involving basic arithmetic and sourced from real news. Thus, we first integrated the GSM8K samples and selectively utilized MetaMathQA samples relevant to variable X to form a new collection of numerical samples, matching the competition's focus on masked numbers. Then, incorporating the analysis of both the NumHG dataset and general mathematical datasets (i.e., GSM8K and MetaMathQA), we have defined all mathematical samples into two categories: addressing mathematical application problems and pure

calculation problems. Specifically, we utilized a large language model's few-shot learning to classify the collected samples. We initially crafted several examples for each problem type as input to guide the mode, which covered a broad spectrum of scenarios to enrich the model's adaptability. Secondly, we instructed the model to distinguish between the given samples, categorizing them as either application or pure calculation problems, as illustrated in Figure 2.

Figure 3 shows examples of mathematical application problems and pure calculation problems. Ultimately, we rely on the model's output to determine the category of the input questions. We found that around 78% of GSM8K questions are application-based, versus 23% in MetaMathQA, aiding our understanding of each dataset's distribution and shaping our data Strategies. Finally, instead of using all external samples, we only retained the numerical-based and application problem samples as supplementary training data. This allows us to maximize the improvement in model performance using as little data as possible.

3.3 Data Self-Augment

Given the high similarity among samples in the NumHG dataset, we try to improve the diversity of samples and the difficulty of the task through data augmentation. Inspired by strategies from the visual domain (Jo and Yu, 2021), we introduced sentence-level random shuffling as a data self-augment strategy, as shown in figure 4. Our goal is to generate structurally diverse training samples while preserving the core information of texts. After reshuffling sentences within each training sample, the LLM continues to perform the original task of filling numerical values across structurally varied texts.

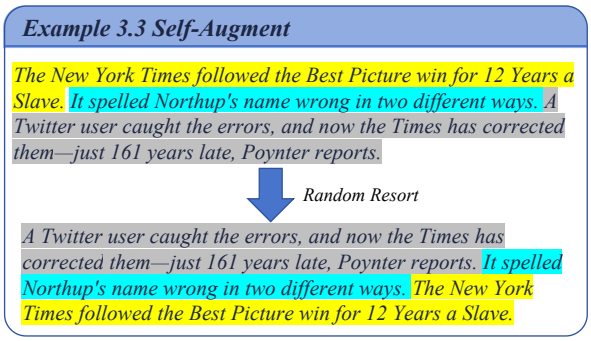

Figure 4: Self-Augment Strategy: sentence-level random shuffling to increase sample diversity while preserving key numerical information.

Although this strategy may disrupt the coherence between sentences within each sample, our experiments have found that it effectively improved the model's ability to handle mathematical problems in more complex contexts.

3.4 Answer Alignment

Adhering to the shared task submission system, the model's output should be a string convertible to a numerical value, devoid of computational methods and descriptive characters. Incorporating the requirements above, we further devised a strategy to simplify the model's output, enhancing model's numerical reasoning performance and shorten the reasoning path. Another reason for implementing this strategy was to ensure data consistency without compromising performance, given the unique characteristics of the competition's computational expressions such as 'Copy and Add'.

Hence, for all samples (i.e., from NumHG, GSM8K and MetaMathQA), we employed regular expressions to directly extract the numerical value as the output for training. Figure 5 shows some example from NumHG. Additionally, this strategy reduced cognitive inference time on 4,921 test instances by eliminating complex computational steps, offering a more direct feedback path.

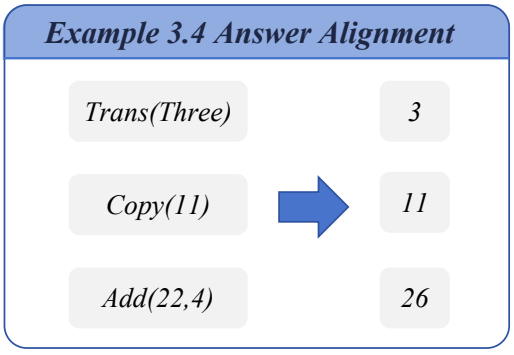

Figure 5: Examples of aligning answers in NumHG, simplifying the model's inference path.

4 Experimental Setup

We utilized the Qwen-72B-Chat model as our base model, on which we performed full parameter fine-tuning. The experiments were conducted on Nvidia A800 GPU with 80GB of memory. During training, we set the maximum token length to 2048, batch size to 8, and performed gradient accumulation every step. An initial learning rate of 5e-6 was set, employing a cosine decay strategy, for a total

of 3 epochs of training. All samples processed by our strategies were used as training data, while the validation set of NumHG were utilized for error analysis. For inference, we employed the default inference parameters.

5 Results

5.1 Final Result

Team	Private Score	Public Score
CTYUN-AI	0.95	0.94

Table 2: Competition results of our team.

At the NumEval-2024 Task 7, the public and private scores were derived from 20% and 80% of the test set data, respectively. In the final standings among all teams, we secured the first place with a private score of 0.95, as shown in Table 2.

This achievement highlights the effectiveness of our method, especially in the more heavily weighted portion of the test set.

5.2 Ablation Study

We took ablation studies to confirm each strategy's (i.e., Data Discovery, Self-Augment, and Answer Alignment) contribution to our final method's success. As shown in Table 3, data discovery, self-augment, and answer alignment brought performance gains of 2%, 4%, 2% separately. Finally, we achieved a 9% performance increase over the baseline in total, underscoring the significance of our approach against high benchmarks.

Method	Private Score	Public Score
Base Data	0.86	0.87
Base Data w/ Prompt	0.87	0.88
+ Data Discovery	0.89	0.88
+ Self-Augment	0.93	0.91
+ Answer Alignment	0.95	0.94

Table 3: Ablation study on our method.

Meanwhile, we evaluated the impact of using samples for mathematical application versus pure calculation problems during data discovery, as shown in Table 4. Findings show application-type problems improve model performance by enhancing real-world numerical understanding, while pure calculation samples negatively affect it due to their complexity leading to intricate computations.

Method Type	Method	Private Score	Public Score
Base Data	w/o Prompt	0.86	0.87
	w/ Prompt	0.87	0.88
w/ Data Discovery	Pure Calculation	0.87	0.87 (-0.01)
	Application	0.89 (+0.02)	0.88

Table 4: Ablation study on data discovery and fusion.

5.3 Error Analysis

In analyzing error cases, we discovered that rounding could lead to misunderstandings in numerical comprehension. For instance, one example states: *"...Nielsen numbers show that 31.1 million people,..."* hence the answer to the question *"____M Watched Jackson Memorial"* should be 31.1, yet the model predicted 31. This indicates that the model's rounding may lead to incorrect answers.

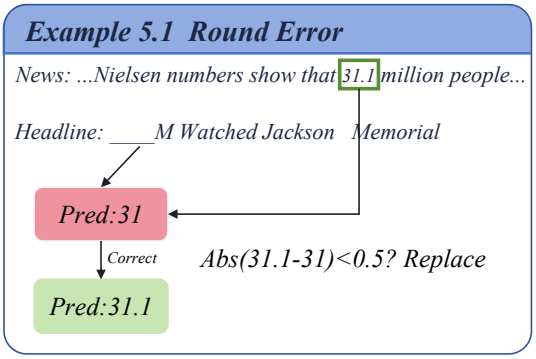

Figure 6: A rounding error case and we introduce a post-processing strategy for it.

As shown in Figure 6, we implemented a post-processing strategy to correct rounding errors. This involves extracting all numbers from the text, comparing them with the model's prediction, and adjusting predictions within a 0.5 difference to the nearest number. This method enhanced our test set performance to 0.95, although it was not included in the competition submission.

6 Conclusion

In this work, we have demonstrated an approach to enhance numerical understanding in large language models (LLMs) using limited data through effective data alignment. Our method integrated data discovery, self-augment, and answer alignment strategies, and significantly improved the model's performance on numerical reasoning tasks. Our success in SemEval-2024 Task 7 highlights the potential of our method in advancing natural language processing, particularly for enhancing the various basic capabilities of large language models.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, and et al. 2023. Qwen technical report. *arXiv:2309.16609*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, and et al. 2020. Language models are few-shot learners.

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

Keith Cortis, André Freitas, Tobias Daudert, Manuela Huerlimann, Manel Zarrouk, Siegfried Handschuh, and Brian Davis. 2017. Semeval-2017 task 5: Fine-grained sentiment analysis on financial microblogs and news. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 519–535.

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2022. Complexity-based prompting for multi-step reasoning. *arXiv preprint arXiv:2210.00720*.

Jian-Tao Huang, Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2023. Numhg: A dataset for number-focused headline generation. *arXiv preprint arXiv:2309.01455*.

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*.

Sanghyun Jo and In-Jae Yu. 2021. Puzzle-cam: Improved localization via matching partial and full features. In *2021 IEEE International Conference on Image Processing (ICIP)*, pages 639–643. IEEE.

Maël Jullien, Marco Valentino, Hannah Frost, Paul O'Regan, Donal Landers, and André Freitas. 2023. Semeval-2023 task 7: Multi-evidence natural language inference for clinical trial data. *arXiv preprint arXiv:2305.02993*.

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. *arXiv preprint arXiv:2308.09583*.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2022. Teaching small language models to reason. *arXiv preprint arXiv:2212.08410*.

Ashutosh Modi, Prathamesh Kalamkar, Saurabh Karn, Aman Tiwari, Abhinav Joshi, Sai Kiran Tanikella, Shouvik Kumar Guha, Sachin Malhan, and Vivek Raghavan. 2023. Semeval 2023 task 6: Legaleval–understanding legal texts. *arXiv preprint arXiv:2304.09548*.

Peiyi Wang, Lei Li, Liang Chen, Feifan Song, Binghuai Lin, Yunbo Cao, Tianyu Liu, and Zhifang Sui. 2023. Making large language models better reasoners with alignment. *arXiv preprint arXiv:2309.02144*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

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

Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Chuanqi Tan, and Chang Zhou. 2023. Scaling relationship on learning mathematical reasoning with large language models. *arXiv preprint arXiv:2308.01825*.