Global Eye 👁: Breaking the "Fixed Thinking Pattern" during the Instruction Expansion Process

Anonymous ACL submission

Abstract

An extensive high-quality instruction dataset is crucial for the instruction tuning process of Large Language Models (LLMs). Recent instruction expansion methods have demonstrated their capability to improve the quality and quantity of existing datasets, by prompting high-performance LLM to generate multiple new instructions from the original ones. However, existing methods focus on constructing multi-perspective prompts (e.g., increasing complexity or difficulty) to expand instructions, overlooking the "Fixed Thinking Pattern" issue of LLMs. This issue arises when repeatedly using the same set of prompts, causing LLMs to rely on a limited set of certain expressions to expand all instructions, potentially compromising the diversity of the final expanded dataset. This paper theoretically analyzes the causes of the "Fixed Thinking Pattern", and corroborates this phenomenon through multi-faceted empirical research. Furthermore, we propose a novel method based on dynamic prompt updating: Global Eye. Specifically, after a fixed number of instruction expansions, we analyze the statistical characteristics of newly generated instructions and then update the prompts. Experimental results show that our method enables Llama3-8B and Llama2-13B to surpass the performance of open-source LLMs and GPT3.5 across various metrics. Our code and data are submitted to the Software & Data option.

1 Introduction

Instruction tuning has emerged as a crucial method for unlocking the remarkable capabilities of Large Language Models (LLMs) (Ouyang et al., 2022; Wei et al., 2022). Instruction tuning is the process of fine-tuning LLMs on an instruction-response dataset to enable LLMs to generate expected responses based on given instructions. The core objective of instruction tuning is to enable LLMs to learn human interaction patterns from the instruction dataset (Zhang et al., 2023). Consequently,

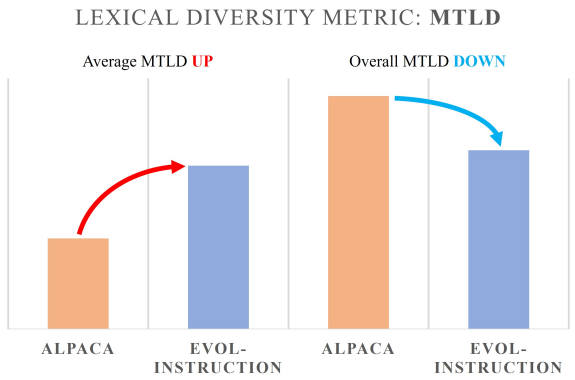

Figure 1: Paradoxical phenomenon of lexical diversity.

the quantity and quality of the instruction dataset become critical factors influencing the effectiveness of instruction fine-tuning on LLMs (Chiang et al., 2023; Zhou et al., 2024).

Recent instruction expansion methods have paved the way for an efficient increase in the quantity and quality (refers to complexity, difficulty, etc) of existing datasets (Xu et al., 2024; Wan et al., 2023; Luo et al., 2023; Zeng et al., 2024). These methods create multi-perspective prompts for high-performance LLMs (e.g., *"Your goal is to create a different instruction from the originals and expanding strategy is [fill in]. Original instruction: [fill in]; New instruction:"*, detailed in A.3). Then those LLMs generate multiple expansion instructions from each original instruction based on those prompts. These methods primarily focus on constructing suitable prompt sets, such as manual construction (Xu et al., 2024; Luo et al., 2023) and automated generation(Zeng et al., 2024). However, once these prompt sets are created, they remain static throughout the instruction expansion process. This results in the repeated use of all prompts and causes LLMs to prefer using a limited set of certain words and expressions to expand all instructions. This phenomenon is called "Fixed Thinking Pattern" in this paper, which harms the diversity of the

new instruction dataset.

This paper begins with a comprehensive investigation of the “Fixed Thinking Pattern" phenomenon and discusses the impact of diversity in instruction datasets on model performance. Furthermore, we propose an innovative solution for this issue. Firstly, we comprehensively investigated existing expansion instruction datasets, focusing on linguistic metrics and word frequency distributions. Surprisingly, our results uncovered a paradoxical phenomenon (shown in Figure 1). While the average lexical diversity of individual samples in expansion datasets showed significant improvement compared to the source dataset, the **overall lexical diversity** of the expansion datasets markedly decreased. We further demonstrate that using the fixed prompt set throughout the expansion process harms the diversity of the final instruction dataset. Finally, our experiments and previous works (Xia et al., 2024; Bukharin and Zhao, 2023) indicate that under the same data distribution, repetitive instructions degrade the diversity of the dataset and negatively influence model performance.

Based on these findings, we propose a novel method: Global Eye, which dynamically updates expansion prompts throughout the instruction expansion process. The fundamental concept of this method is to periodically analyze the word frequency characteristics of newly generated instruction data after a fixed number of expansion steps, and dynamically update the expansion prompts based on these features. This innovative approach effectively compels the LLM to break free from the “Fixed Thinking Pattern" and utilize a richer, more diverse vocabulary in expansion instructions. Experimental results indicate that our method significantly enhances both the individual and overall lexical diversity of the expansion dataset. Further experiments confirmed that the dataset generated by Global Eye enables LLaMA3-8B and LLaMA2-13B to outperform GPT-3.5 across several evaluation metrics.

Our contributions can be summarized as follows:

- We analyze the phenomenon of “Fixed Thinking Patterns" in LLMs’ instruction expansion process, which significantly impacts the diversity of the expansion instruction dataset.
- We propose a framework: Global Eye, which effectively breaks the “Fixed Thinking Pattern" by dynamically updating prompts during the instruction expansion process.
- Experimental results demonstrated that the instruction dataset generated by our method has higher lexical diversity and enabled models to achieve better performance.

2 Fixed Thinking Pattern Phenomenon

2.1 Investigation in Expansion Datasets

We analyze multiple expansion datasets and their source dataset (alpaca), focusing on linguistic metrics and word frequency distribution. To mitigate the influence of low-information words (e.g., ’of’, ’a’, ’an’), we only analyze the characteristics of verbs and nouns. The expansion datasets include evol-instruction (Xu et al., 2024) and Exp-GPT4 & Exp-GPT3.5 made by union prompt set from following works (Wan et al., 2023; Zeng et al., 2024; Xu et al., 2024). All expansion datasets’ quantities are equal (70k).

Index	Alpaca data	Evol-Instruction	Exp-GPT4	Exp-GPT3.5
Average				
Fog	9.82	14.90	17.21	14.78
MTLD	36.08	65.11↑	68.13↑	63.27↑
Overall				
Fog	10.69	15.07	17.09	14.38
MTLD	92.94	71.33↓	87.75↓	72.31↓

Table 1: For Fog and MTLD, higher values indicate greater complexity or diversity.

2.1.1 Linguistic Metrics

We employed the FOG (Goh et al., 2007) to measure the complexity of instruction data and used the MTLD, a length-robust metric (McCarthy and Jarvis, 2010), to assess the lexical diversity of instructions. We analyzed the dataset from two perspectives: **average values** (calculated by averaging the linguistic metrics for each instruction) and **overall values** (computed by combining all instructions into a single text and calculating the linguistic metrics for the entire text). As shown in table 1, the instruction complexity has significantly increased across all levels, which is the primary reason for the effectiveness of the expansion methods. However, the MTLD metric revealed a paradoxical phenomenon: while the average MTLD increased significantly, the overall MTLD decreased. This suggests that LLMs tend to use similar expressions and vocabulary during the expansion process, leading to a decrease in the overall MTLD.

2.1.2 Word Frequency Distributions

The results reveal that in Evol-Instruction, words used more than [1000, 5000, 10000] times account

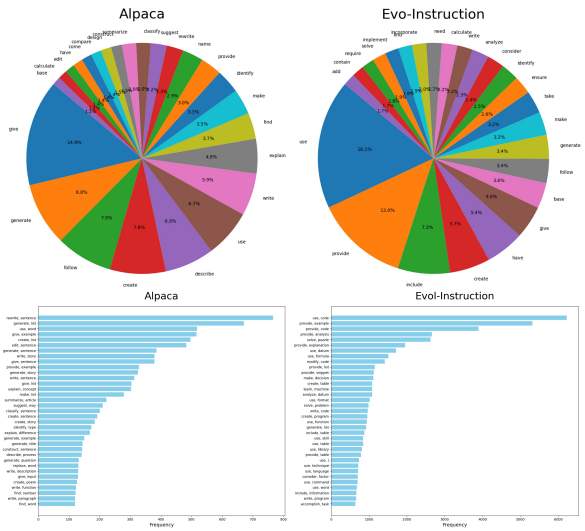

Figure 2: Distributions of words and verb-noun.

for [64.57%, 27.4%, 16.46%] of total word usage. In contrast, these percentages for Alpaca are [33.18%, 8.84%, 0%]. We also analyze the expansion dataset made by a single prompt (detailed setting and **case-by-case** analysis in appendixA.1). About 2,300 words appeared more than five times, accounting for 93% of the total word usage, while 127 words were used over 100 times, making up 59.8% of the total. This demonstrates the extreme word distribution led by existing works. Figure 2 shows the distribution of high-frequency words and verb-noun combinations. It reveals that Evol-Instruction exhibits more polarized distributions of words. That suggests that during the expansion process, LLMs prefer a limited set of specific expression patterns (i.e., particular verb-noun combinations) and words. The other two expansion datasets exhibit similarly, as shown in the appendixA.1.

2.2 Theoretical Analysis

2.2.1 Definitions and Precondition

We define an instruction expansion process that includes M prompts to guide the LLM in generating expansion instructions. The number of instructions to be expanded is N, and the set of all possible words is denoted as V. The set of high-frequency words generated by the t-th prompt is represented as H_t. The instruction expansion method satisfies the following prerequisites:

- **prerequisite 1** Expansion prompts number M of existing methods is always much smaller than the instruction number, with $M \ll N$, where M is always smaller than 10 and N larger than 50k.
- **prerequisite 2** Instructions generated by the same prompt exhibit an uneven word frequency distribution, as seen in section 2.2.

2.2.2 Words Distribution Convergence

As each expansion is independently generated by the same LLM and prompt, these events can be considered independently and identically distributed. Let $X_{i,v}$ be the indicator variable for the occurrence of the word v in the i-th expansion, where $X_{i,v}$ is distributed as a Bernoulli random variable with parameter p_v, $X_{i,v} \sim \text{Bernoulli}(p_v)$. Here, $X_{i,v}$ equals 1 if the i-th expansion contains word v, and 0 otherwise. Define the total occurrence count S_N of word v over N expansions:$S_N = \sum_{i=1}^{N} X_{i,v}$. According to probability theory, the expectation and variance of S_N are given by:

$$\mathbb{E}[S_N] = N \cdot p_v; \text{Var}(S_N) = N \cdot p_v \cdot (1 - p_v)$$

According to the Central Limit Theorem (CLT), when using a single prompt to expand instructions, for the sufficiently large number of expansions N, the sample mean $\frac{S_N}{N}$ converges to the probability p_v, $\frac{S_N}{N} \xrightarrow{a.s.} p_v$, and prompt's words distribution converges to the multinomial distribution, where t means t-th prompt:

$$D_t = Multinomial(p_{v_1}^{(t)}, p_{v_2}^{(t)}, \ldots, p_{v_{|V|}}^{(t)})$$

Since each prompt is used with equal probability during the instruction expansion process, the average occurrence probability P_v and the distribution of words V converges can be expressed as:

$$P_v = \frac{1}{M} \sum_{t=1}^{M} p_v^{(t)}; D = \frac{1}{M} \sum_{t=1}^{M} D_t$$

2.2.3 The Union Words Distribution

Let V be the set of all possible words and the H_t be the set of high-frequency words generated by t-th prompt. The union of the high-frequency word sets H is: $H = \bigcup_{t=1}^{M} H_t$

Assume the size of each H_t is $|H_t| \leq k$. Then the size of the union H satisfies: $|H| \leq M \cdot k$. According to **prerequisite 2**, we know that typically k is much smaller than $|V|$. According to **prerequisite 1**, M is typically a small number, which results in $M \cdot k$ and $|V|$ being of different orders of magnitude.

$$|H| \leq M \cdot k \ll |V|$$

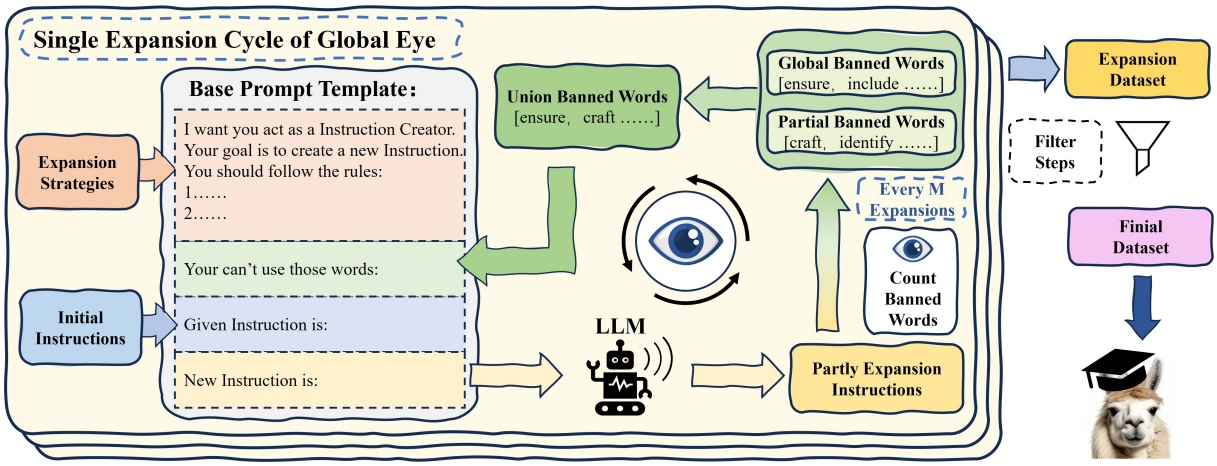

Figure 3: Single expansion cycle of Global Eye. At the beginning of the expansion process, as no words are yet prohibited, the prompt template does not include any prompt regarding forbidden words (as shown in the green section of the figure). Only after first completing the statistics of banned words do we incorporate the relevant prompts into the template.

Assume the total frequency of the high-frequency word set H_t satisfies: $f(H_t) = \sum_{v \in H_t} f(v) > \alpha$, where α is a threshold value. Since H is the union of all H_t sets, the total frequency of H satisfies:

$$f(H) = \frac{1}{M} \sum f(H_t) > \alpha$$

Thus, the union of high-frequency words H is still much **smaller** than the total words and maintains a **high overall frequency**, resulting in D remaining an extremely uneven distribution.

2.2.4 Conclusion

We showed that repeatedly using a single prompt to expand instructions converges to a non-uniform word distribution. Due to the number of reasonable expansion prompts is far less than the number of expansions, the union word distribution remains extreme. This significantly impacts the diversity of the expansion dataset.

2.3 Diversity of the Dataset Influences Model Performance

Previous works(Xia et al., 2024; Bukharin and Zhao, 2023) have shown that removing duplicate instructions from existing datasets significantly enhances the diversity of the datasets and improves model performance. This indicates that increasing the diversity of the dataset positively impacts the final performance of the model. Additionally, Figure 5 demonstrates the relationship between the **overall MTLD values** of the datasets and the model's performance. We constructed datasets of the same quantity and similar complexity (all data having a Fog Index of approximately 19) but with different MTLD values. The results indicate that enhancing the lexical diversity of the dataset can effectively improve the model's final performance.

3 Method: Global Eye

3.1 Motivation

Although previous studies maintained the diversity of datasets by deleting instructions, directly removing instruction data is extravagant for instruction expansion. This practice results in the wastage of substantial access costs associated with LLMs. The goal of Global Eye is to break the "Fixed Thinking Pattern" and enable LLMs to directly produce diverse datasets. We dynamically update all prompts in Global Eye and prevent them from being excessively reused. Specifically, during the instruction expansion process, we analyze high-frequency words in the expansion instructions at fixed steps. We then update all prompts, prohibiting the LLM from using these words in subsequent expansions. This forces the LLM to use new words, thereby breaking its fixed thinking patterns.

3.2 Definition of Instruction Expansion

Assuming we have an initial instruction dataset $D = (I_k, R_k)_{1 \leq k \leq N}$, where I_k is the k-th instruction in D, R_k is the corresponding response for I_k, and number of samples in D is N. In each cycle of instruction expansion, we generate corresponding extension instruction $I^{(t+1)}$ for each I^t in D^t, by prompting LLM with an extension prompt. Then

we use the same LLM to generate responses $R^{(t+1)}$ for $I^{(t+1)}$, which form a new instruction dataset D^t. After T cycles of expansion, we obtain a set of datasets $[D^1, D^2, \ldots D^T]$. Through a final filtering stage, we derive the ultimate expansion instruction dataset D_F, $[D^1, D^2, \ldots D^T] \xrightarrow{filter} D_F$.

3.2.1 Expansion process of Global Eye

In this section, we introduce a **single instruction expansion cycle** of Global Eye, specifically the process that $D^t \rightarrow D^{t+1}$. Figure 3 illustrates this cycle. We first introduce the base prompt template, which consists of four components: (1) Expansion Strategy Prompt, (2) Banned Word Prompt, (3) original instruction input position, and (4) expansion instruction output position. Based on previous work, Global Eye incorporates seven expansion strategies, with an example provided below (see appendix for other strategies). For each instruction expansion, Global Eye first randomly selects a strategy to fill in the expansion strategy prompt. Then, it populates the banned word prompt with the list of banned words. Next, we wrap the initial instruction with this template to prompt the LLM to generate the expansion instruction. After every M instruction expansions, Global Eye compiles an ordered list of high-frequency words from the cumulative expansion instruction dataset, using this to update the banned word list. Notably, at the beginning of the instruction expansion process, the prompt template does not include the banned word prompt as there are no banned words. Only after the first compilation of banned words do we incorporate the banned word prompt into the template. When all instructions in D^t have been expanded, we use the same LLM to generate responses, thereby forming the final expansion instruction dataset D^{t+1}.

Regarding the compilation of banned words, three points are worth emphasizing: (1) These banned words consist of two parts: partially banned words W_p (high-frequency words from the most recent M expansion) and globally banned words W_g (high-frequency words from all expansion instructions). The merging order is as follows: first, the intersection of W_p and W_g, then the remaining words from W_g, and finally the remaining words from W_p. The lengths of W_p and W_g are equal. This approach considers both global high-frequency words while also accounting for local high-frequency words. (2) We only count the frequency of verbs and nouns. This is to avoid the impact of high-frequency words with low information content (such as "of", "a", "an") on the final performance. Additionally, the richness of verbs and nouns to some extent reflects the diversity of expression and vocabulary in the instruction dataset. (3) The number of banned words inserted into the prompt is determined by the union of sets W_p and W_g. We recommend setting the length of both to 4 (rationale provided in the ablation study).

Example of expansion strategy

1. The new prompt that requires only a subset of the skills needed for the Given Prompt described, focusing on a specific area of expertise.
2. Try not to repeat the verb for each instruction in the examples to maximize diversity.
3. The LENGTH and complexity of the new prompt should be similar to that of the Given Prompt.
4. The new prompt must be reasonable and must be understood and responded by humans.

By dynamically updating the Banned Word Prompt, we have achieved the following benefits: (1) A significant increase in the total number of distinct prompts involved in the instruction expansion process; (2) The presence of banned words forces the LLM to use new words and expressions. These effects align with our motivation and contribute to breaking the fixed thinking patterns of the LLM.

3.3 Instruction Data Filter

After T cycles of Global Eye instruction expansion, we obtain a set of datasets $[D^1, D^2, \ldots, D^T]$. We then filter these datasets to form the final dataset D_F. Our filtering process includes the following three rules:

- Instruction keyword filtering: We remove the instruction data containing expansion prompt content such as "New prompt:" or "Given Prompt is:".
- Instruction similarity filtering: We eliminate instructions that are too similar to the original ones. This is done using both rule-based filtering (ROUGE-L score greater than 0.5) and LLM-based filtering.
- Answerability filtering: We have observed that data where the first 20 words of the response contain the keyword "sorry" typically

represent instructions that the LLM cannot answer. Therefore, we remove these data.

After filtering, we form the final dataset D_F.

Prompt of the LLM similarity Filter

Here are two Instructions to GPT4 AI, do you think they are equal to each other, which meet the following requirements:
1. They have same constraints and requirments.
2. They have same depth and breadth of the inquiry.
The First One: {}
The Second one: {}
Your Judgement (Just answer: Equal or Not Equal. No need to explain the reason.):

4 Experiment

4.1 Baselines

Our baseline models include: (1) High-Performance Models: GPT3.5-turbo and GPT4-turbo, the advanced conversational AI models developed by OpenAI, trained on a large corpus of internet text data, demonstrating exceptional natural language interaction capabilities. (2) Open source Model: Alpaca (Taori et al., 2023), a classic open-source LLM built by instruction tuning on the LLaMA model; Vicuna (Chiang et al., 2023), based on the LLaMA model and further fine-tuned on 70,000 user dialogue data, is one of the most advanced and general-purpose open-source instruction-following models; Tulu2 (Wang et al., 2024), the most advanced open-source instruction fine-tuned models; WizardLM (Xu et al., 2024) and Auto-instruct (Zeng et al., 2024) provide extended instruction datasets; UltraLM and UltraLM v2 the powerful open-source models based on large-scale dataset of instructional conversations.

4.2 Experiment Settings

Based on GPT4, we applied Global Eye to perform four rounds of instruction expansion on the alpaca 52k dataset D, resulting in a total of 208k instruction data $[D^1, D^2, D^3, D^4]$. After filtering, we obtained a final set of 145k dataset D_F. The banned word number we set is 4 and we update banned words for every 500 expansion steps. For a fair comparison with the baselines, we randomly sampled 70k instructions from the 145k instructions as our final training set, matching the training data size of the control group settings. Similarly, we have created another dataset containing 70k instruction data **based on GPT-3.5** for comparative experiments. For ease of initialization and comparison, we initialized our model using the LLaMA2 13B base model and LLaMA3 8B base model. We employed the Adam optimizer, with an initial learning rate of 2×10^{-5}. We train our model on A100 GPUs with Deepspeed Zero-3 for 3 epochs, and the batch size is 8 for each GPU.

4.3 Result

4.3.1 Auto Evaluation

To comprehensively evaluate the performance of the Global Eye, we compared our model against baseline models on a series of LLM benchmarks. The benchmarks we used include (1) AlpacaEval and AlpacaEval2 (Dubois et al., 2024), a large language model evaluation framework. (2) MT-bench (Zheng et al., 2024), Using GPT-4 as the evaluator, we score the multi-turn dialogue content generated by LLMs on a scale from 1 to 10. (3)Vicuna-bench (Zheng et al., 2024), Using GPT-4 as the judge, we rate the responses generated by LLMs on a scale of 1 to 10. (4)Wizard eval (Xu et al., 2024), a test set that includes 218 real-world human instructions, we use GPT-4 to assess the quality of responses from LLMs compared to the base model (GPT3.5).

The results are presented in the table 2. Compared to open-source models, Global Eye (GPT-4 based) enabled both Llama2-13B and Llama-8B to achieve outstanding performance, comparable to that of GPT3.5-turbo. Notably, on our dataset, the 8B model outperforms larger models with 13B parameters. This is a remarkable achievement for LLMs of this scale. Furthermore, Global Eye (GPT-3.5 based) also improved the performance of both models, surpassing most open-source models, which further emphasizes the effectiveness and versatility of Global Eye. Compared with other instruction expansion methods, our approach also demonstrated advantages. We also created the Exp-ins dataset, which shares the same expansion process as Global Eye but does not include the dynamic updating prompt. Considering the cost, we employed gpt3.5 to construct this dataset. Compared with Exp-ins, Global Eye(gpt3.5) exhibited significant advantages.

Models	Instruction Num	alpaca eval	alpaca eval2	MT-bench	Vicuna-bench	Wizard eval
High Performance Model						
GPT4-turbo	/	93.6	55.0	9.1	9.1	93.1
GPT3.5-turbo	/	86.3	19.3	7.9	6.7	84.2
Open Source Model						
UltraLM 13B	1500k	80.6	7.1	5.9	5.3	72.5
UltraLM v2 13B	1500k	**86.3**	9.1	6.3	6.7	76.1
Tulu 2 13B	326k	78.9	8.2	6.4	6.2	86.7
Vicuna 13B	70k	70.4	9.2	6.2	5.9	86.9
Model Based on Expansion Dataset and Source Dataset (Alpaca)						
Alpaca 13b	52k	33.25	6.7	4.5	4.7	76.6
WizardLM 13B	70K	75.3	9.8	6.3	6.4	89.1
Auto-Evol 13B	70k	71.2	9.1	6.8	6.6	82.1
Exp-ins(llama2 13b)	70k	73.8	9.2	6.1	6.3	88.1
Exp-ins(llama3 8b)	70k	69.2	8.7	5.9	6.2	81.2
Global Eye Dataset, using GPT4						
Global Eye(llama2 13b)	70k	85.3*	**22.3**	**7.1**	7.7*	**91.7**
Global Eye(llama3 8b)	70k	81.6	20.1*	6.8*	**7.8**	89.9*
Global Eye Dataset, using GPT3.5						
Global Eye(llama2 13b)	70k	79.2	10.2	6.7	6.8	89.4
Global Eye(llama3 8b)	70k	76.3	9.3	6.9	6.5	87.2

Table 2: For fairness, all expansion datasets in the table are derived from the alpaca. The expansion process of **Exp-ins** is consistent with Global Eye, except it does not include the dynamic updating prompt (the banned word part). The highest-performing metrics are highlighted in bold, while the second-highest are marked with an asterisk.

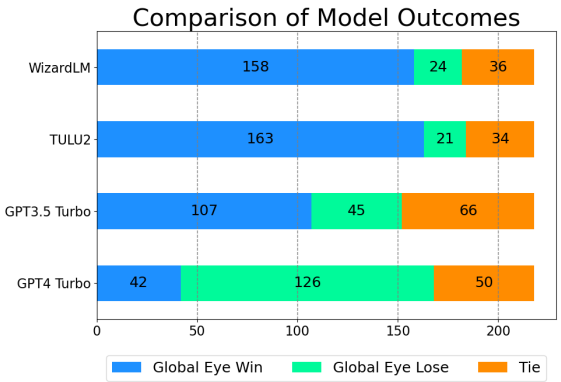

Figure 4: Results of human evaluation.

4.3.2 Human Evaluation

We conducted a human evaluation by using the Wizard Eval test set, which comprises 218 real-world human instructions from various sources. We recruited three highly educated annotators to assess the outputs of our Global Eye (GPT-4 base, Llama2-13B) model against other comparison models. For each annotator, outputs from two models were presented in a randomized order to conceal their origins. The annotators were tasked with judging wins, losses, or ties between pairs of outputs. The final results, as shown in Figure 4, demonstrate that our model has a substantial advantage over open-source models and performs comparably to GPT-3.5. For a model with 13 billion parameters, these results are highly satisfactory. This demonstrates the high quality of instruction data generated by Global Eye.

4.4 Ablation Study

4.4.1 Banned Word Number

In this section, we examine how varying the number of banned words affects the lexical diversity of the final dataset and model performance. We employ GPT-4 to execute Global Eye, setting the number of banned words (both partially banned words and globally banned words) from 1 to 6. Due to cost considerations, we extract a 10k sample from the Alpaca dataset for expansion. We measure the **overall MTLD values** of these datasets and use them to train the Llama2-13B model. The results, as shown in Figure 5, indicate that a banned word count of 4 achieves the optimal balance. This result aligns with our observation of the datasets. When selecting a smaller number of banned words, especially when set to 1, the banned words chosen by Global Eye rotate among only a few terms, such as "ensure" and "craft". However, when the num-

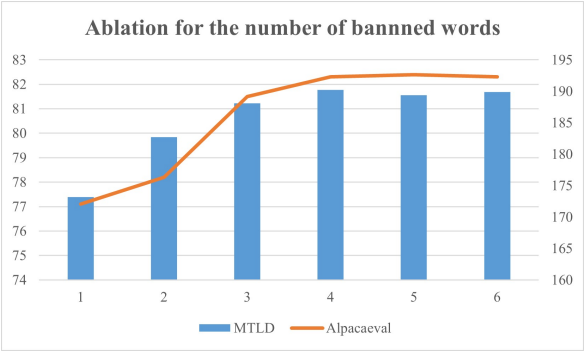

Figure 5: Ablation of the number of banned words. The relationship between overall MTLD and model performance is also observed.

ber of banned words exceeds 3, this situation improves significantly. Conversely, when the number of banned words becomes too large, the frequency of words in the banned word list can no longer be effectively suppressed.

4.4.2 Break the Fixed Thinking Pattern

In this section, we measure the metrics of the datasets produced in the main experiment to verify whether we have successfully broken the LLM's fixed thinking patterns. From the linguistic perspective, as shown in Table 3, all our datasets demonstrate significant improvements in both Fog and MTLD values compared to the original Alpaca dataset, in terms of both average and overall values. Our datasets also maintain a lead in these two indicators compared to Evol-Instruction. More importantly, the expansion by Global Eye not only avoided a decrease in overall diversity but significantly enhanced it.

Figure 6 illustrates the word distribution in the Global Eye (GPT-4 base) dataset. Compared to the distribution in Figure 2, our dataset shows marked improvements in the uniformity of both word and verb-noun combination distributions. Global Eye (GPT-3.5 base) exhibits similar characteristics, with details provided in the appendix. In conclusion, the data generated by Global Eye demonstrates significant advantages in terms of lexical diversity. This stems from its ability to force the LLM to use new words, thereby breaking fixed thinking patterns.

5 Relate Work

5.1 Instruction Expansion

Building a large and high-quality instruction dataset is crucial for instruction tuning. Due to

Index	Alpaca data	Evol-Instruction	Global (GPT4)	Global (GPT3.5)
Average				
Fog	9.82	14.90	19.6	18.9
MTLD	36.08	65.11↑	73.8↑	76.94↑
Overall				
Fog	10.69	14.78	19.32	18.8
MTLD	92.94	71.33↓	190.16↑	97.06↑

Table 3: The last two columns correspond to Global Eye based on GPT-4 and GPT-3.5.

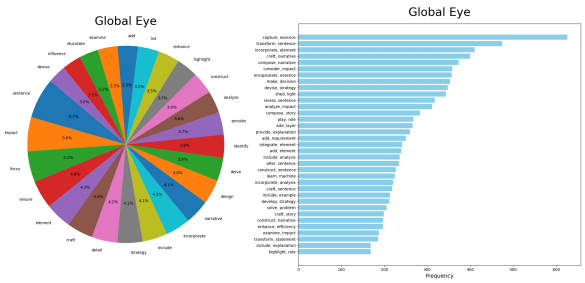

Figure 6: Distributions of words and verb-noun.

the high cost of manually annotating high-quality instruction-response pairs, several studies have advocated for automated data generation processes (Schick and Schütze, 2021; Slack et al., 2023). Leveraging the prompting capability of LLMs (Brown et al., 2020), such as GPT4 (Achiam et al., 2023), (Honovich et al., 2022) prompt LLMs with seed instructions to generate synthetic instructions. These synthetic instructions are then fed into more powerful LLMs (such as ChatGPT) to generate responses for training the target (usually smaller) LLM (Taori et al., 2023). Other studies attempt to expand existing instruction datasets. As a representative work, (Xu et al., 2024; Wan et al., 2023) designed a set of fixed operations to increase instruction complexity and control the difficulty of generated data. Empirical studies by (Wan et al., 2023) further confirm the importance of instruction complexity for LLM alignment.

6 Conclusion

This study first reveals the "Fixed Thinking Pattern" in LLMs during instruction expansion process. Specifically, LLMs tend to favor certain words when expanding instructions, impacting the diversity of expansion instruction datasets. We then analyze the underlying causes of this phenomenon. To address this issue, we propose a dynamic prompt updating solution called Global Eye. This approach compels LLMs to use a richer vocabulary for instruction expansion. Experimental results demonstrate the effectiveness of our method.

Limitation

This paper introduces a method for dynamically updating prompts to expand instruction data: Global Eye. Although this approach significantly enhances the dataset's lexical diversity and the final performance of the model, there is still room for further exploration in the method of updating prompts dynamically. In future work, we plan to extract more features from datasets to provide a more comprehensive perspective for updating prompts.

References

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

T.B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Askell Amanda, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Henighan Tom, Rewon Child, A. Ramesh, DanielM. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, EricJ. Sigler, Mateusz Litwin, Scott Gray, Chess Benjamin, Jack Clark, Christopher Berner, McCandlish Sam, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *arXiv: Computation and Language,arXiv: Computation and Language*.

Alexander Bukharin and Tuo Zhao. 2023. Data diversity matters for robust instruction tuning. *arXiv preprint arXiv:2311.14736*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna. lmsys. org (accessed 14 April 2023)*, 2(3):6.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *Preprint*, arXiv:2404.04475.

Ong Sing Goh, Chun Che Fung, Arnold Depickere, and Kok Wai Wong. 2007. Using gunnnig-fog index to assess instant messages readability from ecas. In *Third International Conference on Natural Computation (ICNC 2007)*, volume 5, pages 480–486.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2022. Unnatural instructions: Tuning language models with (almost) no human labor.

Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. 2023. Wizardcoder: Empowering code large language models with evol-instruct. *arXiv preprint arXiv:2306.08568*.

Philip M McCarthy and Scott Jarvis. 2010. Mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior research methods*, 42(2):381–392.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Timo Schick and Hinrich Schütze. 2021. Generating datasets with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Dylan Slack, Satyapriya Krishna, Himabindu Lakkaraju, and Sameer Singh. 2023. Explaining machine learning models with interactive natural language conversations using talktomodel. *Nature Machine Intelligence*, 5(8):873–883.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6):7.

Fanqi Wan, Xinting Huang, Tao Yang, Xiaojun Quan, Wei Bi, and Shuming Shi. 2023. Explore-instruct: Enhancing domain-specific instruction coverage through active exploration. *arXiv preprint arXiv:2310.09168*.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, et al. 2024. How far can camels go? exploring the state of instruction tuning on open resources. *Advances in Neural Information Processing Systems*, 36:74764–74786.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2024. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

Weihao Zeng, Can Xu, Yingxiu Zhao, Jian-Guang Lou, and Weizhu Chen. 2024. Automatic instruction evolving for large language models. *arXiv preprint arXiv:2406.00770*.

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*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. 2024. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36.

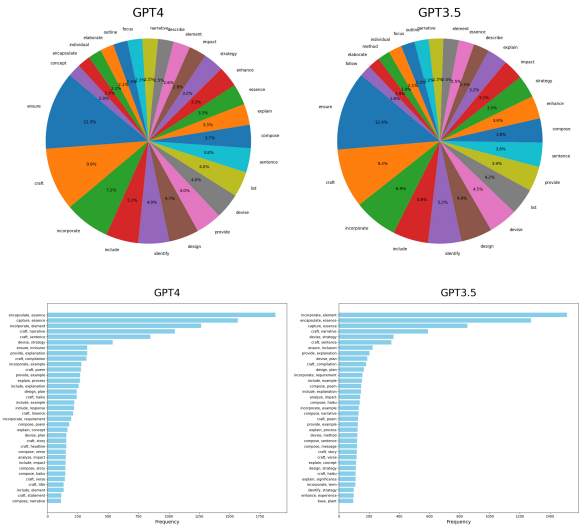

Figure 7: Distributions of words and verb-noun.

A Appendix

A.1 Word Frequency Distributions of Normal Instruction Expansion

In our experiment of the single prompt, we expanded 2,000 original instructions using a single prompt (first prompt in A.3), resulting in a total word count of 143k with 8k unique words. About 2.3k words appeared more than five times, accounting for 93% of the total word count, while 127 words were used more than 100 times, making up 59.8% of the total. We also do the same experiment for other prompts. The word distribution of expansion instruction generated by six out of seven prompts is extremely skewed. The only exception is the prompt that requires the LLM to generate expanded instructions from the original instruction complements. Through detailed case-by-case analysis, we discovered that these prompts add additional conditions, complexity, and difficulty requirements to the instruction expansion. LLMs tend to use common words such as "ensure" and "provide" to impose new constraints on the original instructions, which are the main causes of cognitive fixation. Unfortunately, such prompts are essential in methods of instruction expansion.

Figure 7 illustrates the distribution of high-frequency words and verb-noun combinations. The results indicate that Evol-Instruction demonstrates more polarized distributions in both individual words and verb-noun pairings. This suggests that during the instruction expansion process, LLMs exhibit a preference for specific expression patterns and vocabulary.

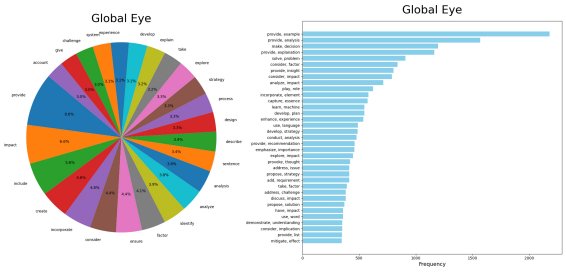

Figure 8: Distributions of words and verb-noun in Global Eye.

A.2 Word Frequency Distributions of Global Eye

Figure 8 depicts the word distribution in the Global Eye (GPT-3.5 base) dataset. In contrast to the distribution shown in Figure 7, our dataset exhibits notably improved uniformity in both word and verb-noun combination distributions. This analysis reveals that the data generated by Global Eye demonstrates superior lexical diversity. This enhancement can be attributed to the method's capacity to compel the LLM to employ a broader vocabulary, effectively disrupting established linguistic patterns.

A.3 Expansion Prompts

Example of prompt

I want you act as a Prompt Rewriter.
Your objective is to rewrite a given prompt into a more complex version to make those famous AI systems (e.g., chatgpt and GPT4) a bit harder to handle.
But the new prompt must be reasonable and must be understood and responded by humans.
Your rewriting cannot omit the non-text parts such as the table and code in Given Prompt. Also, please do not omit the input in Given Prompt.
Please replace general concepts with more specific concepts.
If The Given Prompt contains inquiries about certain issues, the depth and breadth of the inquiry can be increased.
Do not use the following common verbs when constructing prompts: {}
Given Prompt: {}
Directly give me your new prompt without explain ('Created Prompt','New Prompt' are not allowed to appear):

Example of prompt

[breakable] I want you act as a Prompt Rewriter.
Your objective is to rewrite a given prompt into a more complex version to make those famous AI systems (e.g., chatgpt and GPT4) a bit harder to handle.
But the new prompt must be reasonable and must be understood and responded by humans.
Your rewriting cannot omit the non-text parts such as the table and code in Given Prompt. Also, please do not omit the input in Given Prompt.
Please replace general concepts with more specific concepts.
If The Given Prompt contains inquiries about certain issues, the depth and breadth of the inquiry can be increased.
Do not use the following common verbs when constructing prompts: {}
Given Prompt: {}
Directly give me your new prompt without explain ('Created Prompt','New Prompt' are not allowed to appear):

Example of prompt

I want you act as a Prompt Creator.
Your goal is to draw inspiration from the Given Prompt to create a brand new prompt.
This new prompt should belong to the same domain as the Given Prompt but be different and even more rare.
The skills required for the new prompt should be designed to avoid overlapping with the Given Prompt.
The new prompt is complementary to the Given Prompt.
Try not to repeat the verb for each instruction in the examples to maximize diversity.
The LENGTH and complexity of the new prompt should be similar to that of the Given Prompt.
The new prompt must be reasonable and must be understood and responded by humans.
Do not use the following common verbs when constructing prompts: {}
Given Prompt: {}
Directly give me your new prompt without explain ('Created Prompt','New Prompt' are not allowed to appear):

Example of prompt

I want you act as a Prompt Rewriter.
Your objective is to rewrite a given prompt into a more complex version to make those famous AI systems (e.g., chatgpt and GPT4) a bit harder to handle.
But the new prompt must be reasonable and must be understood and responded by humans.
Your rewriting cannot omit the non-text parts such as the table and code in Given Prompt. Also, please do not omit the input in Given Prompt.
Please replace general concepts with more specific concepts.
You should try your best not to make the new prompt become verbose.
Do not use the following common verbs when constructing prompts: {}
Given Prompt: {}
Directly give me your new prompt without explain ('Created Prompt','New Prompt' are not allowed to appear):

Example of prompt

I want you act as a Prompt Creator.
Your goal is to draw inspiration from the Given Prompt to create a fine-grained prompt.
The new prompt that requires only a subset of the skills needed for the Given Prompt described, focusing on a specific area of expertise.
Try not to repeat the verb for each instruction in the examples to maximize diversity.
The LENGTH and complexity of the new prompt should be similar to that of the Given Prompt.
The new prompt must be reasonable and must be understood and responded by humans.
Do not use the following common verbs when constructing prompts: {}
Given Prompt: {}

Directly give me your new prompt without explain ('Created Prompt','New Prompt' are not allowed to appear):

Example of prompt

I want you act as a Prompt Rewriter.
Your objective is to rewrite a given prompt into a more complex version to make those famous AI systems (e.g., chatgpt and GPT4) a bit harder to handle.
But the new prompt must be reasonable and must be understood and responded by humans.
Your rewriting cannot omit the non-text parts such as the table and code in Given Prompt. Also, please do not omit the input in Given Prompt.
If The Given Prompt can be solved with just a few simple thinking processes, you can rewrite it to explicitly request multiple-step reasoning.
You should try your best not to make the new prompt become verbose.
Do not use the following common verbs when constructing prompts: {}
Given Prompt: {}
Directly give me your new prompt without explain ('Created Prompt','New Prompt' are not allowed to appear):

Example of prompt

I want you act as a Prompt Rewriter.
Your objective is to rewrite a given prompt into a more complex version to make those famous AI systems (e.g., chatgpt and GPT4) a bit harder to handle.
But the new prompt must be reasonable and must be understood and responded by humans.
Your rewriting cannot omit the non-text parts such as the table and code in Given Prompt. Also, please do not omit the input in Given Prompt.
You SHOULD add one more constraints/requirements into Given Prompt
You should try your best not to make the new prompt become verbose.
Do not use the following common verbs when constructing prompts: {}
Given Prompt: {}
Directly give me your new prompt without explain ('Created Prompt','New Prompt' are not allowed to appear):