Take Off the Training Wheels! Progressive In-Context Learning for Effective Alignment

Anonymous ACL submission

Abstract

Recent studies have explored the working mechanisms of In-Context Learning (ICL). However, they mainly focus on classification and simple generation tasks, limiting their broader application to more complex generation tasks in practice. To address this gap, we investigate the impact of demonstrations on token representations within the practical alignment tasks. We find that the transformer embeds the task function learned from demonstrations into the separator token representation, which plays an important role in the generation of prior response tokens. Once the prior response tokens are determined, the demonstrations become redundant. Motivated by this finding, we propose an efficient Progressive In-Context Alignment (**PICA**) method consisting of two stages. In the first few-shot stage, the model generates several prior response tokens via standard ICL while concurrently extracting the ICL vector that stores the task function from the separator token representation. In the following zero-shot stage, this ICL vector guides the model to generate responses without further demonstrations. Extensive experiments demonstrate that our PICA not only surpasses vanilla ICL but also achieves comparable performance to other alignment tuning methods. The proposed training-free method reduces the time cost (e.g., 5.45×) with improved alignment performance (e.g., 6.57+). Consequently, our work highlights the application of ICL for alignment and calls for a deeper understanding of ICL for complex generations.

1 Introduction

In-Context Learning (ICL) has attracted growing attention alongside the scaling of Large Language Models (LLMs) (Brown et al., 2020). By conditioning on a handful of input-label pairs as examples, LLMs achieve notable improvements and produce impressive few-shot performance across a range of downstream tasks (Wei et al., 2022). After that, numerous studies have explored the working mechanism of ICL and propose several effective methods to enhance ICL (Hendel et al., 2023; Todd et al., 2023; Wang et al., 2023a; Li et al., 2024).

However, these works mainly focus on classification tasks and simple generation tasks, which limits the exploration of these methods in more complex generation tasks, such as aligning LLMs with human preferences. As a complex and practical task, alignment typically requires training the model, such as Supervise Fine-Tuning (SFT) (Zhou et al., 2023) and Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022). A recent work (Lin et al., 2023) proposed URIAL, a simple method using in-context examples to align several powerful base LLMs and achieves notable instruction-following performance. The success of URIAL demonstrates the feasibility of in-context alignment and encourages us to explore and optimize ICL in the alignment task.

In this paper, we investigate the impact of demonstrations during in-context alignment. We visualize the token distribution KL-divergence of instructions and responses in zero-shot and few-shot settings (Figure 1). To reduce context noise, we set up two few-shot settings with different demonstrations as control groups and have the following observations through comparative experiments: (1) The model likely stores the task function learned from the demonstration in the separator token representation. (2) Demonstrations play a crucial role in prior response generation but are redundant in posterior response generation. These observations highlight the influence of demonstrations on token representation in ICL for alignment tasks, indicating that demonstrations are not always indispensable during the entire response generation stage.

Motivated by these findings, we propose a Progressive In-Context Alignment (**PICA**) method to enhance both the efficiency and effectiveness of regular ICL. Specifically, Our approach involves a two-stage progressive generation strategy: the few-shot stage and the zero-shot stage. During the few-shot stage, the model generates prior part of the response using the standard ICL settings. Subsequently, after generating a specific number of tokens, we transition the model into the zero-shot stage, eliminating the need for further demonstrations to generate the remaining part of the response. To capitalize on the task-related information embedded in the separator tokens, we introduce an ICL vector guidance method. Inspired by the work of task vector in ICL (Hendel et al., 2023; Todd et al., 2023; Li et al., 2024), we extract the ICL vector from the hidden states of specific transformer layers. This vector is then used to steer the model during the zero-shot stage by intervening in the forward pass. PICA minimize the need for demonstrations while improving the quality of generated outputs, thereby reducing the computational cost associated with demonstrations and enhancing overall performance. Extensive experiments show that PICA outperforms regular ICL in both of efficiency and effectiveness. As a training-free method, it is also comparable to other alignment methods (i.e., SFT and RLHF). For example, on average, our PICA boosts the performance of Mistral-7b to reach 90% of the performance of GPT-4-0613. These results support our observations and show the effectiveness of our method in various aspects of alignment. Additionally, we conduct ablation studies to investigate the robustness and generalizability of our method. Our contributions are summarized as follows:

- We delve into the impact of demonstrations on token representation in ICL and qualitatively explore the working mechanism of task functions learned from demonstrations in complex alignment tasks.

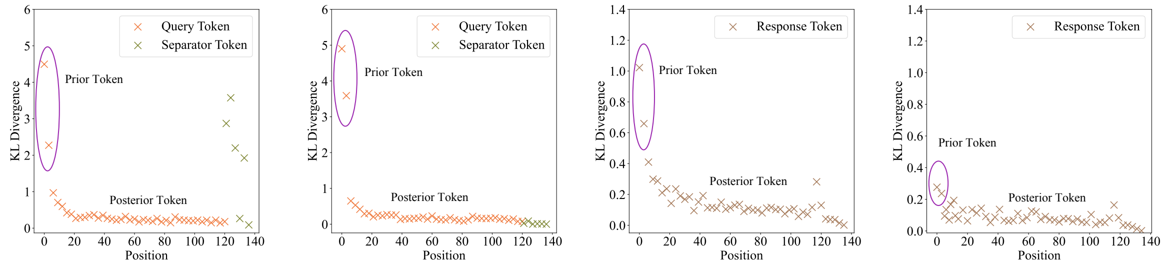
- We propose a progressive in-context alignment method that incorporates progressive generation and ICL vector guidance. This method efficiently aligns models and significantly reduces the computational cost associated with demonstrations.

- We conduct extensive evaluation and ablation experiments on the proposed method, where the results have fully demonstrated its efficiency and effectiveness. Our experiments and analyses provide in-depth insights for future research on in-context alignment.

2 Related Work

LLM Alignment. Prior works have explored alignment tuning through supervised fine-tuning using public instruction datasets (Wang et al., 2022; Zhou et al., 2023; Stiennon et al., 2020) or reinforcement learning from human feedback (Stiennon et al., 2020; Rafailov et al., 2023). A common approach is to fine-tune models using instruction data to enable them to follow instructions effectively. To rapidly accumulate a vast amount of instruction tuning data, Wang et al. (2023b) proposes a pipeline to obtain instruction data from powerful models, such as GPT-4. LIMA leverages only 1000 high-quality instruction data points to fine-tune a 65B parameter LLM (Zhou et al., 2023). It shows that the minimal tuning surprisingly results in a high win rate against ChatGPT. Following instruction fine-tuning, the reinforcement learning is applied to further align the models (Stiennon et al., 2020). Rafailov et al. (2023) introduces a training method for alignment that does not require a reward model. Its powerful convenience and effectiveness have made it one of the *de facto* methods. However, these methods necessitate substantial resources and there is evidence to suggest that such training approaches cause model forgetting of previously acquired knowledge in base LLMs (Wang et al., 2023b; Shen et al., 2023; Wang et al., 2022). In contrast to training-based methods, Lin et al. (2023) experiment with ICL for LLM alignment and Confirm the feasibility of ICL for the alignment task. Building on this finding, we explore a training-free ICL approach. We do not merely utilize ICL. Instead, we initially investigate its working mechanism in token representation learning. This investigation helps enhance the effectiveness of in-context alignment. Similar to us, a very recent concurrent work (Zhan et al., 2024) also identifies the critical role of prior answer token selection in alignment tasks, and proposes a SFT model or external resources guided generation method for multilingual instruction following. Differing from their approach, we focus on the working mechanisms and optimization methods of ICL in the mainstream English alignment tasks.

In-context Learning Working Mechanism. Recent studies have explored the working mechanisms within ICL. Several works try to theoretically demonstrate a strong similarity between the attention patterns in ICL and the process of gradient descent (Akyürek et al., 2023; Dai et al., 2023).



(a) Input Experimental Group (b) Input Control Group (c) Output Experimental Group (d) Output Control Group

Figure 1: The KL-divergence of token probability distributions on Llama2-7b. *Experimental Group* compares zero-shot and few-shot settings, while *Control Group* compares two few-shot settings with different demonstrations. We visualize the input and output separately and mark the prior query tokens and prior response tokens with purple circles.

From a more practical perspective, another line of research suggests that the ICL may function by learning a mapping function from demonstrations, which it then applies to input queries to make predictions (Hendel et al., 2023; Todd et al., 2023; Li et al., 2024). Hendel et al. (2023) extract an ICL task vector from the hidden states and utilize it for intervention during zero-shot inference. Todd et al. (2023) extract a function vector from attention activations using the causal mediation method, which is subsequently added to the hidden states of certain transformer layers during inference. Li et al. (2024) derive a state vector from attention activations and propose several optimization strategies. Unlike these works, we focus on using comparative experiments to explore the impact of demonstrations on token representation, and leverage these findings to enhance the efficiency of ICL.

3 Motivation

In this section, we aim to shed light on the working mechanisms of in-context learning by investigating the following question: **What is the impact of demonstration on token representation in in-context alignment?** To explore this, we design a comparative experiment to highlight how token representations differ between zero-shot and few-shot settings. We use token probability distributions as a proxy for token representations and utilize KL-divergence to measure the shifts in these distributions. By visualizing and quantifying the shifts in token probability distributions caused by demonstrations, we can understand the role of demonstrations in aligning the model and provide further optimization for in-context alignment.

Regarding the experimental setup, we randomly selected 100 data instances of similar length from Ultra-chat (Ding et al., 2023), a commonly used dataset for alignment tuning, as our experimental dataset. For the input prompt, we use a straightforward design by adding several tokens at the end of the query to serve as separator tokens, explicitly distinguishing between the query and the response. We present the visualization results based on the Llama2-7b model in the Figure 1, while the results for other models are provided in Appendix C. We break the token distribution of the whole instance into the input and output parts. A straightforward reason is that the input token distribution shift represents differences in understanding the instruction, while the output token distribution shift represents the ability to respond. By observing and analyzing the visualization, we have two hypotheses: (1) the ICL alignment task function might be encoded into the separator token representation. (2) the quality of response is highly reliant on the quality of prior response tokens.

Input Token Distribution. By comparing the input token probability distributions between zero-shot and few-shot settings, a significant shift is observed in both the prior tokens of the query and the separator tokens. The KL-divergence decreases as the number of query tokens increases. By comparing the experimental group and the control group, we find that the shift in the query distribution also occurs in the control group. However, this shift in the separator tokens is not consistent across different demonstration settings, suggesting distinct underlying causes for these shifts. We attribute the shift in the query's prior token distribution to a "context shift", and we attribute the shift in the separator tokens distribution to a "task shift". Given that LLMs are trained to predict the next token based on the provided context, altering the context directly impacts the token distribution, which we refer to as the "context shift". However, as the number of query tokens increases, the decision space gradually aligns for both zero-shot and few-shot settings, leading to higher consistency in query token prediction and thus a reduced KL-divergence. On the contrary, the trend observed in the query distribution is not mirrored in the separator token distribution. In the control group, the separator token representations remain highly similar. We attribute the large KL-divergence observed in the separator token distribution of the experimental group to the differing tasks, indicating that separator tokens likely encode task-specific information during ICL. We reasonably speculate that the primary impact of demonstration on instruction understanding is reflected in the encoding of separator tokens, where the alignment task function learned through ICL is stored. This hypothesis aligns with prior work (Hendel et al., 2023; Li et al., 2024), yet our findings contribute additional evidence supporting this perspective.

Output Token Distribution. Observing the visualization of output token distribution, we find that when comparing zero-shot and few-shot settings, the response token distribution shows similarity in the posterior tokens. This indicates that the model selects posterior tokens with high consistency in both zero-shot and few-shot settings. When comparing the prior response tokens of the experimental group and the control group, we observe a pattern similar to that of the separator tokens, suggesting that demonstrations play a crucial role in the prior response tokens. Based on these observations and analyses, we speculate that the primary impact of demonstrations on response generation is reflected in the generation of prior answer tokens. Compared to zero-shot settings, demonstrations guide the generation of accurate prior response tokens, which implicitly helps the model successfully follow the instructions. This observation also suggests that once the prior response tokens are determined, the influence of the demonstration diminishes and becomes redundant.

4 Method

Observations from §3 reveal that demonstrations are not always indispensable during the entire response generation stage. To minimize the need for demonstrations while preserving alignment performance, we introduce a progressive in-context alignment approach. This methodology enhances

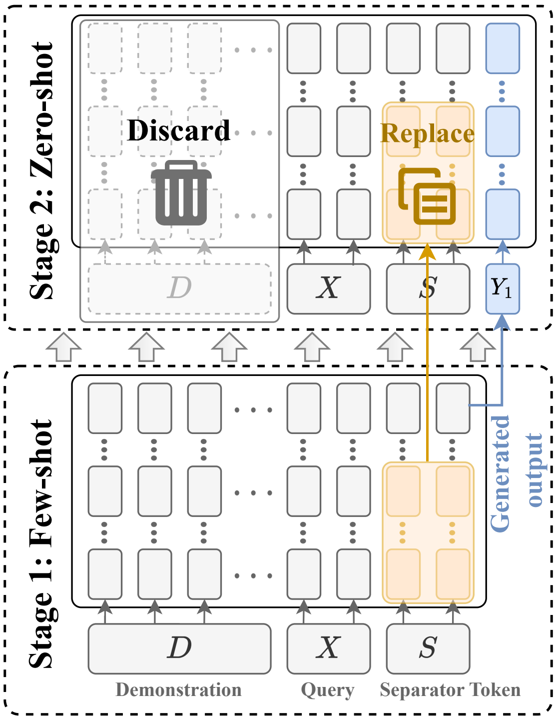


Figure 2: Overview of PICA, which include *few-shot* stage and *zero-shot* stage. The gray block denotes the hidden state and orange block denotes the separator token hidden state that forms the ICL vector. The blue block denotes the generated answer token from few-shot stage.

the efficiency and efficacy of in-context alignment through two innovations: (1) a progressive generation strategy that reduces the computational cost associated with demonstrations, and (2) in-context learning vector guidance that compresses the task function from demonstrations to assist in high-quality response generation.

Inspired by underscoring the redundancy of demonstrations once the pivotal prior response tokens are determined, we introduce a progressive generation strategy, dividing response generation into few-shot and zero-shot stages. During the few-shot stage, the model generates a specific number of prior response tokens by employing a standard in-context learning:

$$Y_i^{\text{few}} = \arg\max_{Y \in V} P(Y|D, Q, S, Y_{1:i-1}^{\text{few}}), \quad (1)$$

where D is the demonstration, Q is the query, S is the separator token, and Y_i^{few} is the i-th answer token generated in few-shot stage. After obtaining several prior answer tokens, the model operates within a more certain and simplified decision space for token generation, allowing the omission of the demonstration to reduce computational costs. Therefore, in the zero-shot stage, the model completes the response based on the existing prior response tokens:

$$Y_i^{\text{zero}} = \arg\max_{Y \in V} P(Y|Q, S, Y_{1:N}^{\text{few}}, Y_{1:i-1}^{\text{zero}}), \quad (2)$$

where N is the number of prior tokens, and Y_i^{zero} the i-th answer token generated in zero-shot stage.

In-context Learning Vector Guidance. Our observations indicate that transformers exhibit task-specific encoding behaviours when encoding the separator token. Recent works (Hendel et al., 2023; Todd et al., 2023) have similar observations, demonstrating that functions learned by ICL can be represented through compressed vectors derived from transformers and can perform simple generation tasks in zero-shot settings. Inspired by these works, we propose the ICL vector guidance to assist the model in generating high-quality responses during the zero-shot stage. Unlike these previous works that intervene single hidden state of the last separator token, we intervene in the initial L layer of all separator tokens. Our preliminary experiments found that this method is more effective for the alignment task, where the output is much longer than that of the simple generation tasks focused on in previous works.

Specifically, during the forward pass in the few-shot generation, we extract the separator token hidden state H_i^{few} from the first L layers, which we combine and refer to as the ICL vector. Subsequently, in the zero-shot stage, we intervene in the separator token representation by replacing the hidden state with the extracted hidden state from the few-shot stage:

$$H_i^{\text{zero}} = \begin{cases} H_i^{\text{few}} & \text{if } i \leq L \\ \text{Layer}(H_{i-1}^{\text{zero}}) & \text{otherwise} \end{cases}, \quad (3)$$

where $\texttt{Layer}(\cdot)$ is the process function of transformer layer. By intervening with the ICL vector, the model receives implicit guidance from the demonstration during generation, thereby improving the quality of the zero-shot stage responses.

Overall, our progressive in-context alignment process is as follows: In the few-shot stage, we utilize standard ICL to generate pivotal prior response tokens while extracting the ICL vector from the separator token representation. Subsequently, we discard the demonstration and employ the ICL vector to guide the model in generating the complete response in the zero-shot setting. This dual-stage progressive in-context alignment approach fully capitalizes on the potential of the ICL vector and the text completion capabilities of foundational language models in the zero-shot setting. By effectively harnessing these capabilities, the approach not only reduces computational cost but also maintains high fidelity in response generation across various settings.

5 Experiment

5.1 Datasets and Models

Recent research demonstrates that utilizing powerful AI assistants such as ChatGPT and GPT-4 for scoring and comparing achieves close alignment with human evaluations while reducing costs (Liu et al., 2023; Dubois et al., 2024). Consequently, we evaluate our method using two automatic alignment benchmarks: alpaca-eval (2.0) (Dubois et al., 2024) and just-eval (Lin et al., 2023). Alpaca-eval comprises 805 instructions and provides a length-controlled win rate from the judge model by comparing the assessed results with those from a reference model. For fast and validated evaluation, we select GPT-3-text-davinci-003 and GPT-4 as reference models, while employing GPT-4-0314 as the judge model. Just-eval includes 800 regular instructions and 200 red-teaming and malicious instructions selected from diverse open-source datasets, offering detailed evaluations across six aspects. On each aspect, scores range from 1 to 5, representing the degree of evaluation. In line with prior work (Lin et al., 2023), we use GPT-4-0314 as the evaluator and report the performance across three random seeds. For efficiency analysis, we evaluate the average inference time on 1000 test data with strictly generated 4096 tokens without using any additional decoding optimization techniques. We report the speedup compared to the standard ICL.

We conduct our experiments using three principal fundamental LLMs: Llama2-7b, Llama2-13b (Touvron et al., 2023) and Mistral-7b (v0.1) (Jiang et al., 2023). These models are selected based on their moderate sizes, open-source availability, and proficiency in ICL. For comparative analysis, we utilized their respective alignment-tuned versions: Llama2-7b-chat, Llama2-13b-chat, and Mistral-7b-Instruct, facilitating a direct comparison with SFT and RLHF. Additionally, our study includes results from OpenAI's GPT models (i.e., GPT-3.5-turbo-0611 and GPT-4-0613), allowing comparison with the state-of-the-art AI assistants. We follow the inference guidelines provided by the authors of these tuned models.

5.2 Implementation Detail

For the in-context learning prompt, we follow previous work (Lin et al., 2023) and use the mainstream system message employed in aligned LLMs. We meticulously designed the demonstrations for in-context learning, creating six examples for alpaca-eval and three examples for just-eval, as they emphasize different evaluation aspects. We utilize greedy generation with a beam size of 1 and set the maximum token length to 4096. The in-context learning vector guidance method we described earlier has a key hyper-parameter, specifically the layer L. Previous studies (Hendel et al., 2023) have demonstrated that the choice of L influences performance. We determine the intervention layer based on the win rate on alpaca-eval. We set the number of prior tokens to 10 as a trade-off between generation quality and efficiency. For consistency and reproducibility, we apply greedy decoding across all experiments. All experiments were conducted on a single NVIDIA A100 80G GPU, with each experiment consuming between 3 to 5 hours of GPU time, depending on the dataset and models used.

5.3 Baseline

In the paper, we compare our method with the following methods and ablation variants:

- **SFT or RLHF** is the baseline with alignment tuning method. We strictly follow the guidelines provided by the creators of these tuned models during inference.
- **Zero-shot** is the baseline for the zero-shot setting that uses only the given query as input, and **Vanilla ICL** is the regular ICL which makes predictions on the label by taking both demonstration and instruction.
- **Vec.** is the ablation variants that only utilize ICL vector guidance in zero-shot setting, while **Prog.** is the ablation variants that apply progressive generation strategy without ICL vector guidance during zero-shot stage.

5.4 Main result

Table 1 presents the win rates of each baseline on alpaca-eval and the scores on just-eval, as well as the speedup for efficiency analysis. In addition to our complete PICA method, we also present evaluation results for two ablation variants (i.e., 'Vec.' and 'Prog.') to explore the effectiveness of the two proposed innovations. The combination of these innovations constitutes our PICA method.

PICA outperforms the baseline with tuning-free baselines. As shown in the Table 1, our method outperforms zero-shot and vanilla ICL baselines across three models on alpaca-eval. On the just-eval dataset, our PICA also surpasses the tuning-free baseline in the majority of aspects. Compared to regular ICL, our method effectively improves helpfulness, factuality, engagement, and safety. However, in terms of clarity and depth, our method shows a minor decline. We attribute this to the fact that our approach still has limitations in generating consistently information-rich responses, indicating that the ICL vector cannot fully encapsulate all the information provided by the demonstration.

PICA is comparable to the alignment tuning methods. When compared to SFT or RLHF models, our approach demonstrates superior performance on the alpaca-eval dataset, indicating an overall advantage over SFT and RLHF methods. However, on the just-eval dataset, the results vary across different aspects. For instance, in the aspects of helpfulness and factuality, our method excels, highlighting its capability to follow instructions and generate high-quality and accurate responses. This also supports the widespread hypothesis that alignment tuning may cause models to forget some of their knowledge (Wang et al., 2023b; Shen et al., 2023). Conversely, in terms of clarity, depth, and engagement, our method lags slightly, suggesting that SFT and RLHF have an advantage in producing high-quality response styles over ICL. In terms of safety, our method surpasses SFT but does not exceed RLHF, indicating that ICL provides relatively basic safety alignment. On the other hand, with strong models such as Llama2-13b or Mistral-7b, the performance of our PICA can reach 90% of the performance of GPT-3.5 and GPT-4.

PICA achieves high efficiency compared to vanilla ICL. Analyzing the speedup shown in Table 1, our method significantly reduces the time cost compared to vanilla ICL (e.g., achieving a 5.45× speedup on Llama2-7b) and is close to the zero-shot method across three models. This improvement is attributed to our progressive generation strategy, which successfully saves a substantial amount of time by discarding the demonstration. Notably, our method is orthogonal to attention speedup techniques, such as flash attention (Dao et al., 2022) and page attention (Kwon et al., 2023).

Models + Alignment Methods	Alpaca-eval		Just-eval						Speedup
	vs GPT-3	vs GPT-4	Helpful	Clear	Factual	Deep	Engaging	Safe	
GPT-3.5-turbo-0611	69.51	46.46	4.82	4.97	4.84	4.33	4.66	4.99	-
GPT-4-0613	72.51	53.52	4.86	4.99	4.90	4.49	4.61	4.97	-
Llama2-7b-chat (RLHF)	40.50	17.49	4.12	**4.84**	4.13	**4.18**	**4.77**	**5.00**	5.68
Llama2-7b (Zero-shot)	24.65	11.74	2.78	3.01	3.11	2.27	2.29	1.05	5.81
Llama2-7b (Vanilla ICL)	42.47	15.00	4.01	4.10	4.16	3.50	3.31	1.98	1.00
Llama2-7b (Vec.)	36.51	13.73	3.68	3.72	3.80	3.01	2.94	1.73	5.43
Llama2-7b (Prog.)	42.13	16.23	3.78	3.82	3.94	3.26	3.04	1.78	5.53
Llama2-7b (PICA)	**45.90**	**21.57**	**4.21**	4.09	**4.30**	3.41	3.42	2.09	5.45
Llama2-13b-chat (RLHF)	55.30	38.60	4.36	**4.94**	4.36	**4.55**	**4.83**	**5.00**	4.97
Llama2-13b (Zero-shot)	33.73	15.20	3.26	3.65	3.60	2.63	2.62	1.86	5.31
Llama2-13b (Vanilla ICL)	59.82	37.61	4.38	4.70	**4.68**	4.37	4.24	4.09	1.00
Llama2-13b (Vec.)	53.57	24.43	4.24	4.45	4.24	3.85	3.79	2.22	4.84
Llama2-13b (Prog.)	58.14	34.91	4.25	4.33	4.35	3.60	3.48	4.01	4.78
Llama2-13b (PICA)	**62.78**	**40.15**	**4.58**	4.66	**4.68**	4.16	4.15	4.37	4.83
Mistral-7b-instruct (SFT)	62.78	43.30	4.72	4.75	4.30	4.41	4.37	2.00	4.95
Mistral-7b (Zero-shot)	43.32	22.55	3.86	4.14	4.05	3.38	3.31	1.61	5.23
Mistral-7b (Vanilla ICL)	62.03	40.35	4.70	**4.87**	**4.81**	4.32	4.38	3.03	1.00
Mistral-7b (Vec.)	61.19	37.61	4.76	4.81	4.74	4.36	4.32	2.48	5.02
Mistral-7b (Prog.)	62.75	39.73	4.76	4.84	4.77	**4.42**	**4.61**	4.17	4.83
Mistral-7b (PICA)	**66.38**	**44.33**	**4.79**	4.86	4.79	**4.42**	4.59	**4.34**	4.93

Table 1: Comparison of Alignment Performance and Efficiency. Alpaca-eval presents the win rate against competitor models, while just-eval presents the scores across six aspects (scores are on a scale of 1-5). Results highlighted in gray represent our methods: *Vec.* denotes the ICL vector guidance and *Prog.* denotes progressive generation ablation variants. The best results in each aspect are marked in **bold**. Speedup indicates the efficiency improvement compared to vanilla ICL.

We will leave further exploration for future work.

Both progressive generation strategy and ICL vector guidance contribute to performance improvement. We conduct ablation experiments on our proposed progressive generation strategy and ICL vector guidance, as indicated by the results highlighted in grey in Table 1. When only one of these methods is used, the model's performance declines, with a more significant drop observed when the progressive generation strategy is removed. This clearly demonstrates the effectiveness of both methods, with the progressive generation strategy playing a more critical role. It also indicates the limitations of ICL vector guidance, which, while effective in simpler tasks (Hendel et al., 2023; Todd et al., 2023), shows constraints in more complex alignment tasks.

Overall, our method outperforms ICL in performance and efficiency, achieving results comparable to alignment tuning. These promising outcomes validate the effectiveness of our approach and empirically support our understanding of the role of demonstrations in in-context alignment.

6 Analysis

6.1 Layer Selection

We delve into the impact of layer selection on the extraction of the ICL vector. We evaluate the performance based on the win rate compared to GPT-3-text-davinci-003 on the alpaca-eval datasets, as shown in Figure 3. Our results reveal a dual-phase trend: initially, increasing the number of layers improves performance, but this improvement stops or slightly declines in the later layers. This indicates that the ICL function is dynamically stored within the separator token representation. In the initial layers, transformers primarily focus on learning and encapsulating the ICL function within the hidden state, where additional layers enhance the richness of the functional information in the ICL vector. In contrast, the later layers prioritize applying this learned information for prediction tasks. Here, additional layers tend to introduce noise, causing a slight drop in performance. This also suggests that

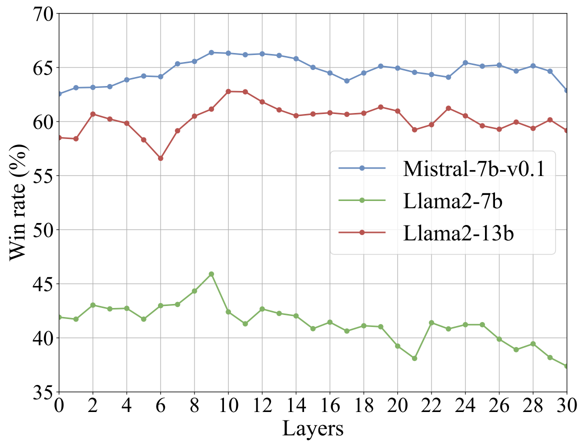


Figure 3: Win rate comparing with GPT-3-text-davinci-003 on alpaca-eval for each choice of the intermediate layer L.

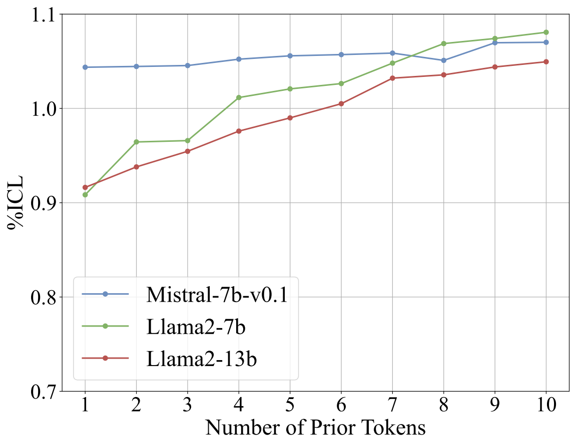


Figure 4: Win rate comparing with GPT-3-text-davinci-003 on alpaca-eval for number of the prior token on three models. We normalize the result with vanilla ICL result.

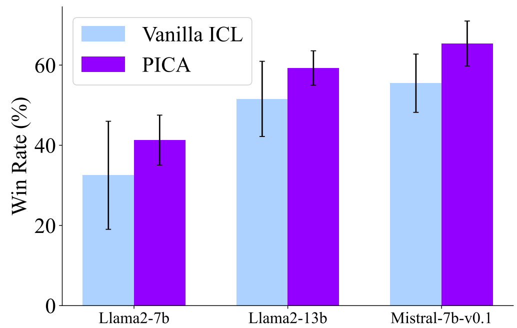


Figure 5: The mean and standard error of ICL and PICA performance with five demonstration across three models.

our method is not significantly affected by layer selection, confirming the robustness of our approach.

6.2 Prior token Ablation

Figure 4 presents an ablation study on the number of prior tokens across three models, normalized by the vanilla ICL results. An intuitive conclusion is that increasing the number of prior tokens improves the model's performance, and with about 8 prior tokens, PICA surpasses vanilla ICL. However, this improvement trend gradually diminishes. When the number of prior tokens reaches 10, the performance gain becomes less significant. This indicates that the demonstration aligns approximately the first 10 tokens to human performance. After generating 10 tokens, the base model can largely complete the response generation independently.

6.3 Robustness Analysis

In this section, we examine the robustness of PICA to demonstration selection. Specifically, we evaluate the performance of ICL and PICA across three models using five different sets of demonstrations. The results, including the mean and standard deviation of the performance metrics, are shown in Figure 5. We observe that the ICL method is more sensitive to changes in the demonstrations compared to the PICA method across all three models. This indicates that PICA effectively enhances robustness. We attribute this to our approach of explicitly incorporating demonstrations only in the prior response tokens, while using implicit demonstration representations during the zero-shot generation stage. This strategy effectively mitigates the impact of suboptimal demonstrations on performance.

Winner	Ratio (%)
Mistral-7b (**PICA**)	35.4
Mistral-7b-instruct (SFT)	24.1
Tie	40.5
Llama2-13b (**PICA**)	34.6
Llama2–13b-chat (RLHF)	21.3
Tie	44.1

Table 2: Results of human evaluation: The win rate of pairwise comparisons between PICA and SFT or RLHF.

6.4 Human Evaluation

We randomly sampled 100 examples each from the alpaca-eval and just-eval datasets, presenting the responses generated by PICA alongside those from the SFT or RLHF models to computer science graduate students who serve as annotators. We asked the annotators to choose which response was better or if it was a tie. Table 2 shows the results, which align with the automated evaluation.

7 Conclusion

In this paper, we investigate and analyze the impact of demonstrations on token representation in in-context alignment through comparative experiments. Based on our observations and analyses, we introduce a novel progressive in-context alignment method that significantly reduces the need for demonstrations while preserving alignment performance. Extensive experiments indicate that PICA outperforms tuning-free baselines in both effectiveness and efficiency, achieving performance that is better or comparable to SFT or RLHF. Our experiments and analyses provide in-depth insights for future research on ICL in alignment. In the future, we aim to further explore the mechanisms and optimizations of ICL in more complex tasks.

Limitations

Despite our discoveries and improvements, we must acknowledge certain limitations in our work:

Model Size: We evaluated our method on Llama2-7b, Llama2-13b, and Mistral-7b, and these experiments were conducted on a limited scale with moderately sized models. This limits our exploration of the application of PICA on larger models. We will explore the use of PICA on larger models such as llama2-70b in future work.

Theoretical Foundation: Our conclusions about the role of demonstration and ICL working mechanism lack rigorous theoretical grounding. In exploring the working mechanism of ICL, we derived some hypotheses through comparative experiments on token representation. While these hypotheses provided insights, they lack solid mathematical derivation and a theoretical basis, limiting the generalizability of our method. For example, in Appendix B, we analyze a kind of instruction that PICA does not handle well.

Evaluation Datasets: Most of our experiments utilized the alpaca-eval and just-eval datasets, which are based on AI assistant automated evaluation pipelines. Related work (Dubois et al., 2024) has shown that these GPT-4-based evaluation methods can introduce biases, such as a preference for longer responses, which may affect the accuracy of our experimental results. Additionally, our dataset quantity is still limited, and the evaluation metrics do not fully cover all aspects of alignment, such as mathematics, reasoning, and coding. We will continue to explore our method with more comprehensive evaluation metrics in future work.

References

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. 2023. What learning algorithm is in-context learning? investigations with linear models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. 2023. Why can GPT learn in-context? language models secretly perform gradient descent as meta-optimizers. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 4005–4019. Association for Computational Linguistics.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. 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*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 3029–3051. Association for Computational Linguistics.

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

Roee Hendel, Mor Geva, and Amir Globerson. 2023. In-context learning creates task vectors. In *EMNLP (Findings)*, pages 9318–9333. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *CoRR*, abs/2310.06825.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles, SOSP 2023, Koblenz, Germany, October 23-26, 2023*, pages 611–626. ACM.

Dongfang Li, Zhenyu Liu, Xinshuo Hu, Zetian Sun, Baotian Hu, and Min Zhang. 2024. In-context learning state vector with inner and momentum optimization. *CoRR*, abs/2404.11225.

Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. 2023. The unlocking spell on base llms: Rethinking alignment via in-context learning. *CoRR*, abs/2312.01552.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *EMNLP*, pages 2511–2522. 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, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Lingfeng Shen, Sihao Chen, Linfeng Song, Lifeng Jin, Baolin Peng, Haitao Mi, Daniel Khashabi, and Dong Yu. 2023. The trickle-down impact of reward (in-)consistency on RLHF. *CoRR*, abs/2309.16155.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. 2020. Learning to summarize from human feedback. *CoRR*, abs/2009.01325.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. 2023. Function vectors in large language models. *CoRR*, abs/2310.15213.

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.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023a. Label words are anchors: An information flow perspective for understanding in-context learning. In *EMNLP*, pages 9840–9855. Association for Computational Linguistics.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. 2023b. How far can camels go? exploring the state of instruction tuning on open resources. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-naturalinstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 5085–5109. Association for Computational Linguistics.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Trans. Mach. Learn. Res.*, 2022.

Runzhe Zhan, Xinyi Yang, Derek F. Wong, Lidia S. Chao, and Yue Zhang. 2024. Prefix text as a yarn: Eliciting non-english alignment in foundation language model. *CoRR*, abs/2404.16766.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. LIMA: less is more for alignment. In *NeurIPS*.

A Case Study

We present a case study comparing SFT, ICL, and PICA on Mistral-7b in Figure 6. The SFT model incorrectly stated that Canada was colonized by the British in 1607, leading to poor performance in factuality with a score of 1. This highlights a common issue with SFT models, where they may forget acquired knowledge over time. As a result, the SFT model received low marks in helpfulness (3) and engagement (2), despite a reasonable clarity score (4). This misrepresentation shows the limitations of the SFT approach in retaining and accurately recalling historical facts. The ICL model is relatively better in factuality. However, the generated content lacked depth and richness, scoring 2 in depth and 2 in helpfulness, suggesting that while the ICL method generates some stylistic tokens, it does not produce sufficiently detailed or useful responses. Our PICA model provided a comprehensive and accurate response, detailing the colonization history of Canada, resulting in high scores across all aspects: helpfulness (5), clarity (5), factuality (5), depth (4), and engagement (4). The PICA model effectively combined stylistic tokens with detailed and accurate information, showcasing its capability to generate high-quality responses that are both informative and engaging.

B Error Analysis

In our preliminary experiments, we found that the proposed PICA approach frequently performed poorly in generating enumeration-type responses (e.g. "Give me a list of some famous world music artists."). Consequently, we analyzed the KL-divergence of responses to these instructions in zero-shot and few-shot settings. The visualization results are shown in Figure 7. Our observations indicate that, although the trend of KL-divergence is generally similar to what we observed in §3 there are differences in each enumeration of the response. We found that the KL-divergence of prior tokens is usually larger than the posterior tokens in each enumeration, indicating that these prior enumeration tokens are pivotal. The quality of responses to enumerative instructions is influenced not only by the selection of prior response tokens but also by the selection of prior enumeration tokens. We attribute this to the fact that each enumerated item is relatively independent of each other. When generating these enumerations, the model requires more substantial guidance from the demonstrations. However, the proposed ICL vector and the positions of previous enumeration responses do not provide enough information for generation, thus reducing the quality of each enumeration. This highlights a limitation of our current PICA approach, which we will explore and optimize in future work.

C More Exploration on Demonstration

We present additional comparative experiments to further delve into the impact of demonstrations on token representation. We conduct experiments on Llama2-7b and Mistral-7b models with the same data as in §3. The experimental group includes both zero-shot and few-shot methods, while the control group includes two different demonstrations in few-shot settings. In addition to the KL-divergence of token distributions, we introduce two new metrics for measuring the difference between the two methods, i.e., Top Token Prob and Top Token Rank.

Top Token Rank refers to the ranking position of a token predicted by one method within the token distribution of another method. Specifically, given the context, we first obtain the next predicted token from one method and then determine its rank within the token distribution of the other method. A lower Top Token Rank manifests a greater overlap in the decision space under the greedy search setting. Analogously, Top Token Prob indicates the probability of a token predicted by one method within the distribution of another method. In this case, we first obtain the next predicted token from one method and then report its probability in the token distribution of the other method. A higher Top Token Prob manifests a higher likelihood of obtaining the same result under the greedy sampling setting. These metrics directly reflect the similarity between the generation results of the two methods. By evaluating both Top Token Rank and Top Token Prob, we can obtain a comprehensive understanding of how closely the methods align in terms of their token predictions.

The KL-divergence results on Mistral-7b are shown in Figure 8, and we find a similar pattern to that in Figure 1.

The Top Token Rank results are shown in Figure 9 and Figure 10. For convenience, we set the rank of the token to 10 if it is greater than 10. We find that tokens with a large KL-divergence difference typically also have a higher Top Token Rank, indicating that our understanding of demonstrations applies to this observation as well. Notably,

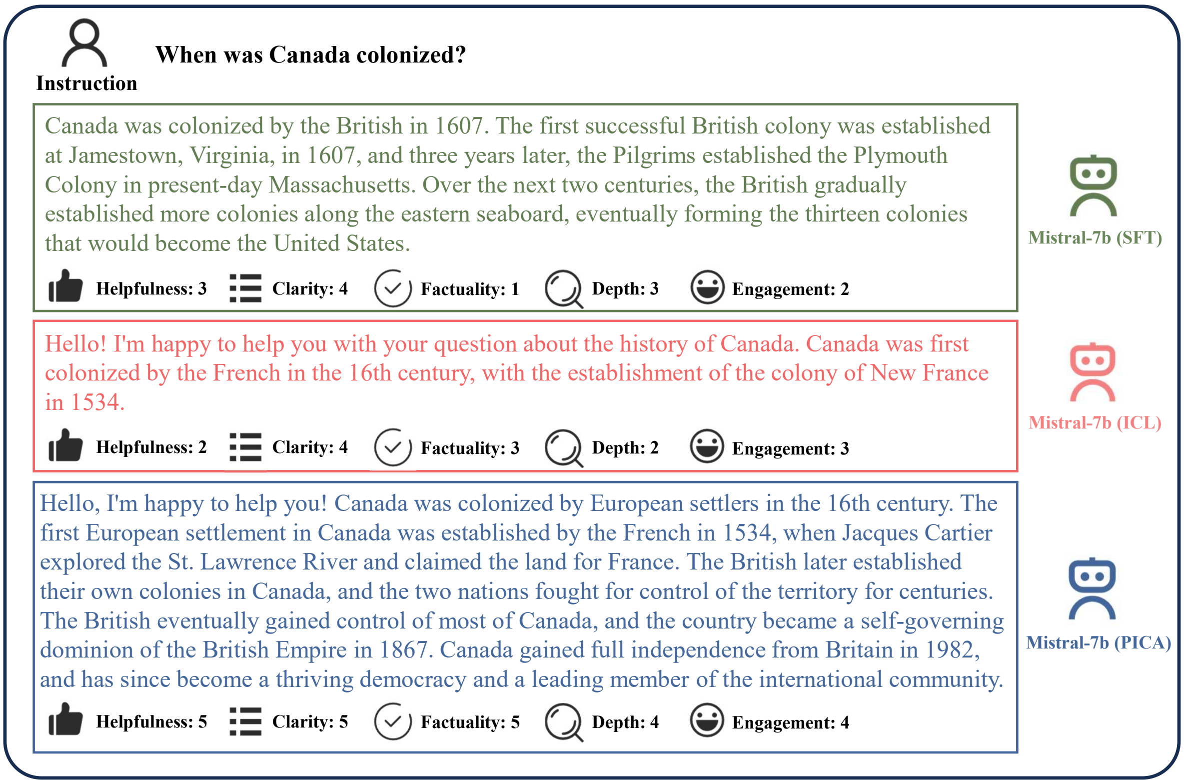
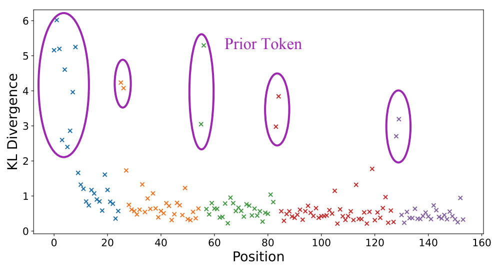
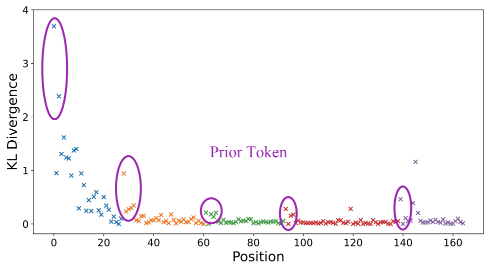


Figure 6: Case study of SFT, ICL, and PICA on Mistral-7b. We report results of the five regular evaluation aspects on just-eval.



(a) Llama2-7b



(b) Mistral-7b

Figure 7: KL-divergence of response token distributions of enumerative instructions on Llama2-7b and Mistral-7b.

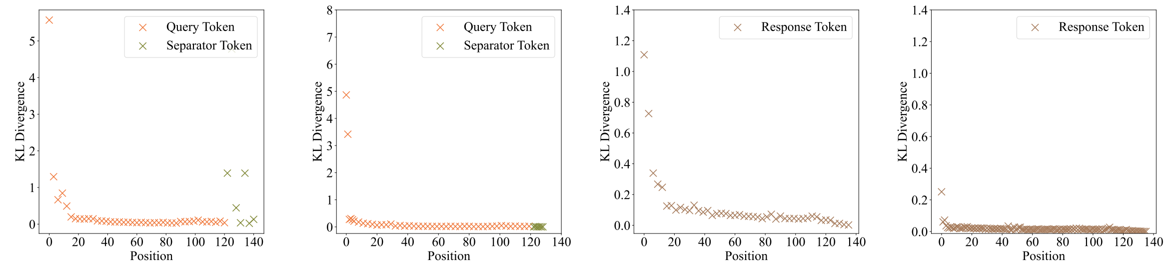
even though the separator token distribution differs significantly, the Top Token Rank remains low. This observation suggests that though demonstrations have a lot of influence on the separator token representation, the predicted next token rank still remains unchanged.

The Top Token Prob results are shown in Figure 11 and Figure 12, where we find that tokens with a large KL-divergence difference typically also have a low Top Token Prob. This further supports our understanding of the role that demonstrations play in the ICL. Similar to the result of Top Token Rank, the predicted separator token probability is high, indicating that demonstration will not change the selection of separator token.

Overall, we observe similar patterns across KL-Divergence, Top Token Rank, and Top Token Prob metrics, despite minor differences. This demonstrates the generalizability and universality of our understanding of the impact of demonstrations.

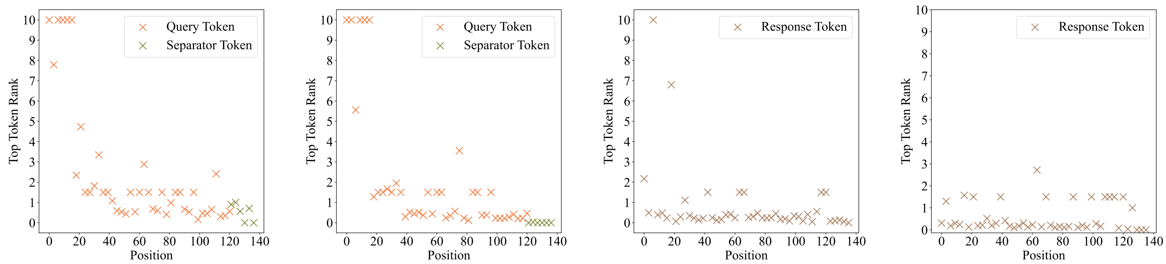
D PICA Prompt

We present the default version prompt with one example used in our experiment in the Table 3.



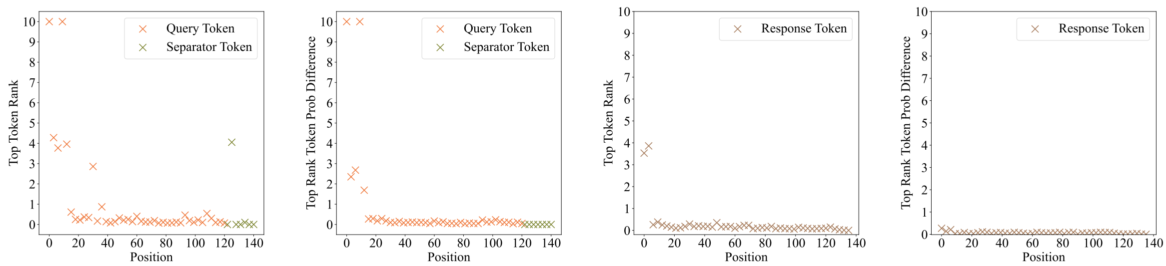
(a) Input Experimental Group (b) Input Control Group (c) Output Experimental Group (d) Output Control Group

Figure 8: The KL-divergence of token probability distributions on Mistral-7b. *Experimental Group* compares zero-shot and few-shot settings, while *Control Group* compares two few-shot settings with different demonstrations. We visualize the input and output separately and mark the prior query tokens and prior response tokens with purple circles.



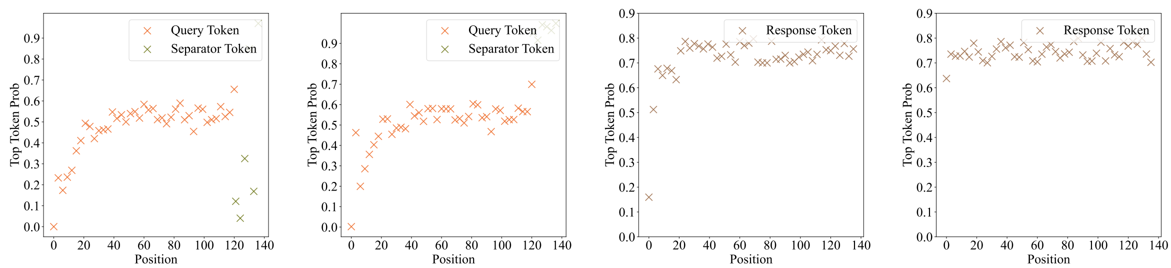
(a) Input Experimental Group (b) Input Control Group (c) Output Experimental Group (d) Output Control Group

Figure 9: Average Top Token Rank on Llama2-7b. *Experimental Group* compares zero-shot and few-shot settings, while *Control Group* compares two few-shot settings with different demonstrations. We visualize the input and output separately



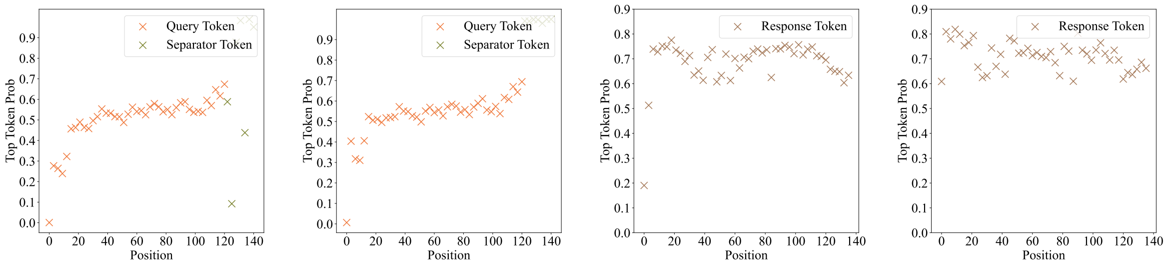
(a) Input Experimental Group (b) Input Control Group (c) Output Experimental Group (d) Output Control Group

Figure 10: Average Top Token Rank on Mistral-7b. *Experimental Group* compares zero-shot and few-shot settings, while *Control Group* compares two few-shot settings with different demonstrations. We visualize the input and output separately



(a) Input Experimental Group (b) Input Control Group (c) Output Experimental Group (d) Output Control Group

Figure 11: Average Top Token Prob on Llama2-7b. *Experimental Group* compares zero-shot and few-shot settings, while *Control Group* compares two few-shot settings with different demonstrations. We visualize the input and output separately



(a) Input Experimental Group (b) Input Control Group (c) Output Experimental Group (d) Output Control Group

Figure 12: Average Top Token Prob on Mistral-7b. *Experimental Group* compares zero-shot and few-shot settings, while *Control Group* compares two few-shot settings with different demonstrations. We visualize the input and output separately

The default version of PICA prompt with an example

[System Message]
Instruction

Below is a list of conversations between a human and an AI assistant (you).
As an AI assistant, you will engage in conversations with users, responding to their queries which are presented under the heading "# Query:".
Your responses should be entered under the heading "# Answer:".
You excel in a wide range of tasks including, but not limited to, providing general information, conducting reasoning, engaging in role-play, creative writing, planning, and solving mathematical and coding problems.
Your responses should be well-structured, comprehensive, and aim to thoroughly address the user's query or problem at hand.
When enumerating items in your responses, limit the examples to no more than ten, and avoid completely redundant content.
Please ensure that your responses are encapsulated within triple backticks ("```") at the start and end to maintain formatting consistency throughout the conversation.

Query:
```
Look for poems that mention the given object or character. The names of the poems and their poets should be written.

the moon
```

Answer:
```
These are seven examples of poems that mention the moon.
1. "The Moon and the Yew Tree" by Sylvia Plath
2. "The Moon" by Robert Louis Stevenson
3. "Above the Dock" by T. E. Hulme
4. "Moonrise" by D. H. Lawrence
5. "The Freedom of the Moon" by Robert Frost
6. "To the Moon" by Percy Bysshe Shelley
7. "The Cat and the Moon" by W. B. Yeats

This celestial body is a frequent motif in poetry, symbolizing various themes such as beauty, mystery, and the passage of time, making it a beloved subject for many poets.
```

Table 3: The default version of PICA prompt with an example