Focused Large Language Models are Stable Many-Shot Learners

Anonymous ACL submission

Abstract

In-Context Learning (ICL) enables large language models (LLMs) to achieve rapid task adaptation by learning from demonstrations. With the increase in available context length of LLMs, recent experiments have shown that the performance of ICL does not necessarily scale well in many-shot (demonstration) settings. We theoretically and experimentally confirm that the reason lies in more demonstrations dispersing the model attention from the query, hindering its understanding of key content. Inspired by how humans learn from examples, we propose a training-free method FocusICL, which conducts triviality filtering to avoid attention being diverted by unimportant contents at token-level and operates hierarchical attention to further ensure sufficient attention towards current query at demonstration-level. We also design an efficient hyperparameter searching strategy for FocusICL based on model perplexity of demonstrations. Comprehensive experiments validate that FocusICL achieves an average performance improvement of 5.2% over vanilla ICL and scales well with many-shot demonstrations.

1 Introduction

The rapid development of large language models (LLMs) has facilitated the emergence and enhancement of their In-Context Learning (ICL) abilities (Wei et al., 2022; Dong et al., 2023). As a training-free method, ICL can achieve fast model adaptation on specific tasks based on several demonstrations prefixed to the query, formally denoted as $\mathtt{ICL}(response|demos, query)$. Intuitively, more demonstrations can help LLMs better understand the task and increase the likelihood of finding demonstrations that aid in responding queries, thus leading to better performance. Theoretically, a similar conclusion can be drawn. Previous studies (Dai et al., 2023; Irie et al., 2022; von Oswald et al., 2023; Akyürek et al., 2023) have theoretically inferred that ICL can be viewed as an implicit

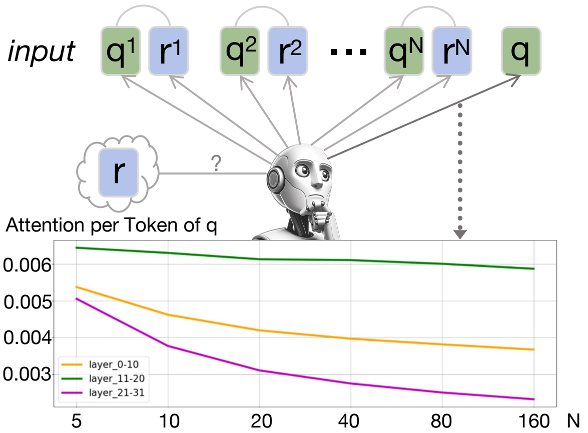


Figure 1: The average model attention for query is dispersed by the increased number of demonstrations, causing inadequate understanding of query.

finetuning process, with demonstrations analogous to training samples. On this basis, as finetuning has been validated to comply with the scaling law (Hernandez et al., 2021) where performance increases with the number of training samples, the performance of ICL should also positively correlates with the number of demonstrations, which has been experimentally verified by previous studies (Bertsch et al., 2024; Duan et al., 2023).

However, with the increase in available context length of LLMs (Reid et al., 2024), some studies (Zhao et al., 2023; Agarwal et al., 2024) observe counterexamples when scaling the demonstration numbers from few-shot to many-shot. Agarwal et al. (2024) finds that the optimal number of demonstrations for six out of eleven benchmarks is not the maximum number they have tested. Our experimental results (Figure 5) also indicate that the model performance might decline with increased demonstrations when applying ICL, exhibiting an inverse-scaling phenomenon (McKenzie et al., 2023). These findings indicate that LLMs are not stable many-shot learners.

To understand this gap, we revisit the derivation of Dai et al. (2023) that formally equates ICL with finetuning and identify that their approximation of standard attention operation as linear attention operation will ignore the competition for attention between demonstrations and the query when generating the response. Since this approximation is key to the equivalence of ICL and finetuning, we hypothesize that the reason why ICL does not adhere to the scaling law like finetuning is that more demonstrations can divert attention away from the query. Inadequate attention and understanding of the query can naturally lead to inferior response. To verify our hypothesis, we first conduct experiments confirming that increasing the number of demonstrations does lead to a decrease in model attention towards queries (Figure 1). We further experiment by adding blank spaces within the demonstrations and confirm that: the more blank spaces added, the more attention towards queries distracted by blanks, resulting in lower response accuracy (Figure 2).

Inspired by the way humans benefit from ignoring irrelevant contents and integrating insights from multiple examples when solving problems, we propose FocusICL to avoid the attention dispersion issue faced by ICL. Specifically, at the token-level, FocusICL conducts triviality filtering by adaptively masking unimportant tokens of demonstrations based on attention distribution, allocating the attention to more important contents. At the demonstration-level, FocusICL performs hierarchical attention mechanism by dividing demonstrations into multiple batches and respectively conducting intra-batch and inter-batch attention operations. The limited demonstration number within each batch ensures sufficient attention to the query, while inter-batch attention integrates the benefits from a larger number of demonstrations. We further introduce an efficient hyperparameter searching strategy for FocusICL according to model perplexity of demonstrations.

Our experiments across three LLMs on five benchmarks confirm that FocusICL achieves an average performance improvement of 5.2% over ICL by avoiding attention dispersion, with lower inference overhead. This demonstrates the effectiveness, efficiency, and generalizability of FocusICL. Furthermore, we observe that FocusICL achieves performance scaling with the number of demonstrations by maintaining attention on critical parts, making demonstration number a possible scaling direction for LLM-based AGI. Finally, we propose a unified perspective to understand the divergent phenomena observed in previous studies, where more demonstrations lead to either improved (Bertsch et al., 2024) or deteriorated (Agarwal et al., 2024) performance in ICL. Based on experimental results, we conclude that the performance of ICL initially benefits but subsequently suffers from more demonstrations. The weaker the model and the closer the relationship between samples, the later the sweet spot for the number of demonstrations occurs.

Our contributions are summarized as follows:

1. We analyze that the reason more demonstrations may lead to a decline in ICL performance is that they degrade the model understanding of query by dispersing its attention.
2. We propose FocusICL to achieve rational attention allocation via triviality filtering operation and hierarchical attention mechanism, making LLMs stable many-shot learners.
3. We conduct comprehensive experiments and analyses to validate the effectiveness, efficiency, generalizability and scalability of FocusICL.

2 Background

Formalization of ICL We follow (Dong et al., 2023) to define the general ICL paradigm. Given an LLM $\boldsymbol{\mathcal{M}}$ and a query $\boldsymbol{q}$, we choose N demonstrations from a candidate set $\boldsymbol{\mathcal{S}_{demos}} = \{(\boldsymbol{q_i}, \boldsymbol{r_i})\}_{i=1}^{M}$ to attain the response $\boldsymbol{r}$ from $\boldsymbol{\mathcal{M}}$ as follows:

$$\boldsymbol{r} = \texttt{Sampling}(\boldsymbol{\mathcal{M}}(\texttt{Cat}[\underbrace{\boldsymbol{q_0}; \boldsymbol{r_0}; ...; \boldsymbol{q_N}; \boldsymbol{r_N}}_{\boldsymbol{demos}}; \boldsymbol{q}])) \quad (1)$$

where $\texttt{Sampling}(\cdot)$ denotes certain sampling strategy and $\texttt{Cat}[\cdot]$ denotes the operation of concatenation.

Scaling Demonstration Number Due to restrictions on context window (2048 $\sim$ 4096), early studies (Brown et al., 2020; Lu et al., 2022) on ICL are limited to few-shot scenarios where they generally observe gains from more demonstrations. As the context window expands recently, counterexamples occur. Agarwal et al. (2024) finds that the best performance of Gemini 1.5 Pro is achieved under settings where demonstration number is not the maximum one tested in over half of the benchmarks. Zhao et al. (2023) discoveries that increasing the number of demonstrations does not necessarily improve model performance across five LLMs. We observe similar phenomena in Figure 5.

3 Revisiting

In this section, we explore what impedes LLMs from becoming stable many-shot learners.

3.1 Approximating ICL as Finetuning

Since Dai et al. (2023) derives that ICL is formally equivalent to finetuning, with demonstrations analogous to training samples, we decide to revisit their derivation process below to explore why finetuning satisfies scaling laws (Hernandez et al., 2021) while ICL does not.

Finetuning Let $\boldsymbol{W}_0, \boldsymbol{\Delta W}_{FT} \in \mathbb{R}^{d_{out} \times d_{in}}$ be the initialized parameter matrix and the update matrix, and $\boldsymbol{x} \in \mathbb{R}^{d_{in}}$ be the input representation. The output of certain linear layer optimized by gradient descent can be formulated as follows:

$$\hat{\boldsymbol{x}} = \boldsymbol{x}\boldsymbol{W}_0 + \boldsymbol{x}\boldsymbol{\Delta W}_{FT} \tag{2}$$

ICL For each attention head of $\boldsymbol{\mathcal{M}}$, let $\boldsymbol{h}_i \in \mathbb{R}^{d_{in}}$ be the representation of the ith input token, $\boldsymbol{W}_q, \boldsymbol{W}_k, \boldsymbol{W}_v$ be the projection matrices for computing the queries, keys and values. We denote $\boldsymbol{h}_{i \in \boldsymbol{demos}}\boldsymbol{W}_k$, $\boldsymbol{h}_{i \in \boldsymbol{demos}}\boldsymbol{W}_v$, $\boldsymbol{h}_{i \in \boldsymbol{q}}\boldsymbol{W}_k$, $\boldsymbol{h}_{i \in \boldsymbol{q}}\boldsymbol{W}_v$ as $\boldsymbol{D}_k$, $\boldsymbol{D}_v$, $\boldsymbol{Q}_k$, $\boldsymbol{Q}_v$, respectively. To generate $\boldsymbol{r}$, the output of $\boldsymbol{h}_{\boldsymbol{r}}$ can be derived below:

$$\begin{aligned}
&\hat{\boldsymbol{h}}_{\boldsymbol{r}} \\
&= \mathrm{Att}(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q, \mathrm{Cat}[\boldsymbol{D}_k; \boldsymbol{Q}_k], \mathrm{Cat}[\boldsymbol{D}_v; \boldsymbol{Q}_v]) \\
&\approx \mathrm{LinAtt}(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q, \mathrm{Cat}[\boldsymbol{D}_k; \boldsymbol{Q}_k], \mathrm{Cat}[\boldsymbol{D}_v; \boldsymbol{Q}_v]) \\
&= \boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q \,\mathrm{Cat}[\boldsymbol{D}_k; \boldsymbol{Q}_k]^\top \begin{bmatrix} \boldsymbol{D}_v \\ \boldsymbol{Q}_v \end{bmatrix} \\
&= \boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{Q}_v\boldsymbol{Q}_k^\top + \boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{D}_v\boldsymbol{D}_k^\top \\
&= \boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_{ZSL} + \boldsymbol{h}_{\boldsymbol{r}}\Delta\boldsymbol{W}_{ICL}
\end{aligned} \tag{3}$$

Dai et al. (2023) approximate the standard attention to linear attention by removing the `softmax` operation for ease of qualitative analysis. Since $\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{Q}_v\boldsymbol{Q}_k^\top$ is the attention result in the zero-shot learning (ZSL) setting and $\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{D}_v\boldsymbol{D}_k^\top$ is the extra outcome from demonstrations, they are denoted as $\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_{ZSL}$ and $\boldsymbol{h}_{\boldsymbol{r}}\Delta\boldsymbol{W}_{ICL}$ respectively. Comparing Eq. (3) with Eq. (2), we can understand ICL as finetuning by treating the $\Delta\boldsymbol{W}_{ICL}$ generated from demonstrations as the $\Delta\boldsymbol{W}_{FT}$ generated from training samples.

3.2 Ignorance of Attention Competition

From Eq. (3) we can further derive as follows:

$$\begin{aligned}
&\hat{\boldsymbol{h}}_{\boldsymbol{r}} \\
&\approx \underbrace{\mathrm{LinAtt}\left(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q, \boldsymbol{Q}_k, \boldsymbol{Q}_v\right)}_{\text{outcome from } \boldsymbol{q}} + \underbrace{\mathrm{LinAtt}(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q, \boldsymbol{D}_k, \boldsymbol{D}_v)}_{\text{outcome from } \boldsymbol{demos}}
\end{aligned} \tag{4}$$

which means that the existence of demonstrations does not affect the outcome from q. However, when we no longer approximate standard attention as linear attention, we arrive at the opposite conclusion:

$$\begin{aligned}
&\hat{\boldsymbol{h}}_{\boldsymbol{r}} \\
&= \mathrm{Att}(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q, \mathrm{Cat}[\boldsymbol{D}_k; \boldsymbol{Q}_k], \mathrm{Cat}[\boldsymbol{D}_v; \boldsymbol{Q}_v]) \\
&= \mathrm{softmax}(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q \,\mathrm{Cat}[\boldsymbol{D}_k; \boldsymbol{Q}_k]^\top) \begin{bmatrix} \boldsymbol{D}_v \\ \boldsymbol{Q}_v \end{bmatrix} \\
&= (1 - \lambda(\boldsymbol{h}_{\boldsymbol{r}}))\, \mathrm{softmax}(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{Q}_k^\top)\boldsymbol{Q}_v \\
&\quad + \lambda(\boldsymbol{h}_{\boldsymbol{r}})\, \mathrm{softmax}(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{D}_k^\top)\boldsymbol{D}_v \\
&= (1 - \lambda(\boldsymbol{h}_{\boldsymbol{r}}))\, \underbrace{\mathrm{Att}\left(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q, \boldsymbol{Q}_k, \boldsymbol{Q}_v\right)}_{\text{outcome from } \boldsymbol{q}} \\
&\quad + \lambda(\boldsymbol{h}_{\boldsymbol{r}})\, \underbrace{\mathrm{Att}\left(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q, \boldsymbol{D}_k, \boldsymbol{D}_v\right)}_{\text{outcome from } \boldsymbol{demos}},
\end{aligned} \tag{5}$$

where:

$$\lambda(\boldsymbol{h}_{\boldsymbol{r}}) = \frac{\sum_i \exp\left(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{D}_k^\top\right)_i}{\sum_i \exp\left(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{D}_k^\top\right)_i + \sum_j \exp\left(\boldsymbol{h}_{\boldsymbol{r}}\boldsymbol{W}_q\boldsymbol{Q}_k^\top\right)_j} \tag{6}$$

With the existence of $\lambda(\boldsymbol{h}_{\boldsymbol{r}})$ in Eq. (5), an increase in the number of demonstrations will lead to a larger $\lambda(\boldsymbol{h}_{\boldsymbol{r}})$, thereby decreasing the model attention towards q. At the same time, ICL does not necessarily adhere to the scaling law as it is no longer formally equivalent to finetuning. **Therefore, we hypothesize that more demonstrations can divert model attention from the key contents (query), leading to possible performance decrease.**

3.3 Experimental Evidence for Hypothesis

To validate our hypothesis, we first investigate whether the model attention towards the query decreases with the increase of demonstration number. To avoid potentially unreliable results caused by data contamination (Jiang et al., 2024), our exploratory experiments are conducted with longchat-

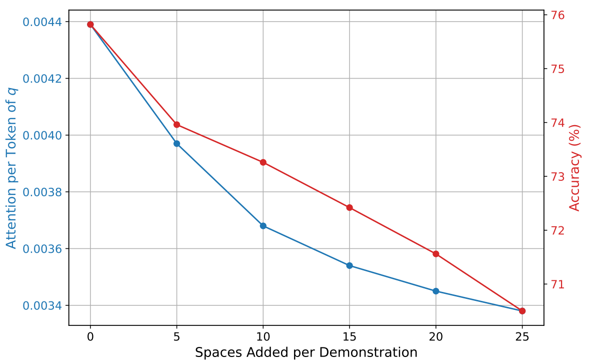


Figure 2: Accuracy and attention of LONGCHAT-7B-V1.5-32K with varying number of spaces added per demonstration. Demonstration number is set as 100.

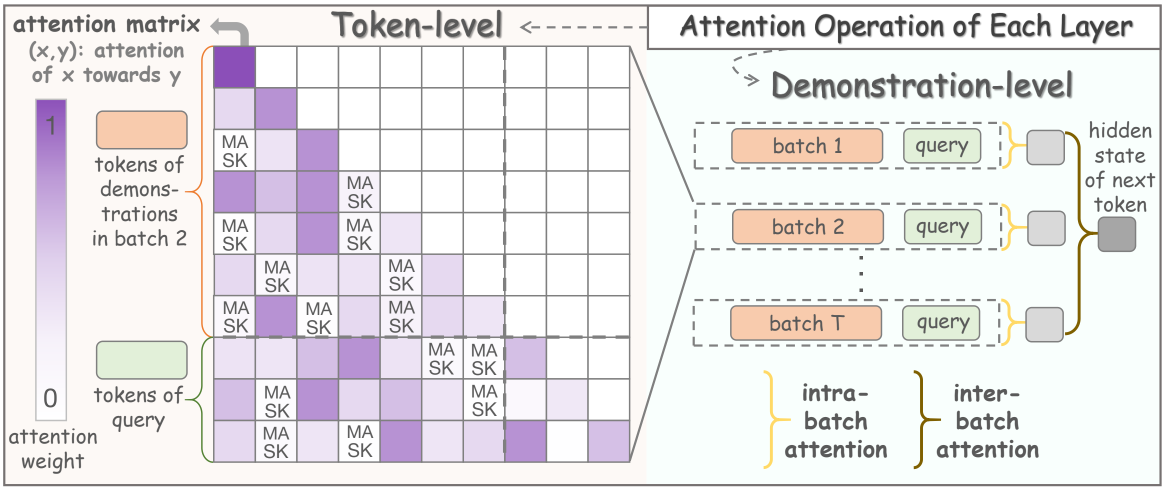


Figure 3: Overall illustration of FocusICL.

7b-v1.5 (Dacheng Li* and Zhang, 2023) (32k context window) on the proposed CountA benchmark (See details in §), which requires the model to **Count** the number of character '**A**' in the five candidates. As shown in Figure 1, the average attention weight of model towards each token in the query decreases by scaling up the demonstration number, corresponding to Eq. (5).

We further explore how the model's lack of attention towards the query affects the quality of the response. Specifically, we add several blank spaces at the end of each demonstration. This format maintains the ICL paradigm and the meaningless blank spaces will not introduce additional information. As shown in Figure 2, we find that more blank spaces disperse the model attention towards the query similar to the demonstrations, which in turn leads to a decline in accuracy. Based on the experiments above, we have confirmed our hypothesis.

4 Methodology

To mitigate the impact of LLMs' attention being dispersed by many-shot demonstrations, we propose FocusICL. The core idea behind FocusICL is to allocate model attention to more important contents at token-level by triviality filtering (§4.1) and at demonstration-level by hierarchical attention (§4.2), as shown in Figure 3.

4.1 Triviality Filtering

Humans benefit from selectively ignoring irrelevant parts (trivialities) of demonstrations to avoid attention dispersion. In contrast, the standard attention mechanism of LLMs fails to completely ignore (assign zero attention weight to) trivialities and leverage the prior that the tokens of query are generally important, for which we propose triviality filtering operation. To predict response $\boldsymbol{r}$ for given query $\boldsymbol{q}$, in each attention layer, we first calculate the attention scores $\boldsymbol{s}$ as follows:

$$\boldsymbol{s} = \boldsymbol{h_r} \boldsymbol{W}_q \,\mathrm{Cat}[\boldsymbol{D}_k; \boldsymbol{Q}_k]^\top \tag{7}$$

Instead of directly applying `softmax` on $\boldsymbol{s}$ like standard attention operation, we filter the trivialities in the demonstrations according to a pre-set threshold p in advance as follows:

$$\begin{aligned}
&\texttt{index} = \texttt{arg}\{\texttt{index}|\texttt{count}(\boldsymbol{s} \leq \boldsymbol{s}_{\texttt{index}}) = p \times |\boldsymbol{s}|\} \\
&\texttt{mask}(\boldsymbol{s}) = \begin{cases} -\texttt{INF}, \ \boldsymbol{s}_i \leq \boldsymbol{s}_{\texttt{index}} \ \texttt{and} \ i \in \boldsymbol{demos} \\ 0, \ \texttt{else} \end{cases} \\
&\hat{\boldsymbol{h}}_{\boldsymbol{r}} = \texttt{softmax}(\boldsymbol{s} + \texttt{mask}(\boldsymbol{s}))\,\mathrm{Cat}[\boldsymbol{D}_v; \boldsymbol{Q}_v]
\end{aligned} \tag{8}$$

where $\hat{\boldsymbol{h}}_{\boldsymbol{r}}$ is the outcome of $\boldsymbol{h_r}$. By applying triviality filtering operation, useless parts of demonstrations are assigned zero attention weights thus LLMs can focus on leveraging relevant contents of the demonstrations to solve the current query. To achieve a broad impact, apart from $\boldsymbol{r}$, we also apply triviality filtering operation on tokens belong to responses of demonstrations by autoregressively treating $\{(\boldsymbol{q}_i, \boldsymbol{r}_i)\}_{i=1}^{k-1}$ as demonstrations of $(\boldsymbol{q}_k, \boldsymbol{r}_k), k \in [2, N]$.

4.2 Hierarchical Attention

When there are numerous examples, humans draw inspirations for problem-solving from different examples separately and then integrate the insights to avoid distracting attention by focusing on too many examples simultaneously. Motivated by this, we introduce hierarchical attention mechanism for LLMs

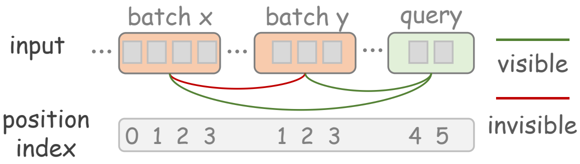


Figure 4: Input details of FocusICL.

to learn from many-shot demonstrations while focusing on current query. We first split the demonstrations into T batches, where each one comprises B consecutive demonstrations. Without editing the token order, we change the position indexes to ensure that each batch is logically adjacent to the query (Figure 4). To ensure that batches are mutually invisible to each other, we use a mask matrix, allowing us to parallelly apply intra-batch attention within each batch i and query as follows:

$$\hat{\boldsymbol{h}}_{\boldsymbol{r}}^{\boldsymbol{i}},\ \boldsymbol{s}^{\boldsymbol{i}} = \texttt{TrivialityFiltering Att}(\boldsymbol{h}_{j\in batch_i \cup \boldsymbol{q}}) \tag{9}$$

By controlling the batch size B, we can ensure that the model maintains enough attention towards the query within each batch. To further integrate insights from different batches, we conduct inter-batch attention as follows:

$$\hat{\boldsymbol{h}}_{\boldsymbol{r}} = \sum_{i=1}^{T} \hat{\boldsymbol{h}}_{\boldsymbol{r}}^{\boldsymbol{i}} \times \frac{\sum_j e^{s_j^i}}{\sum_k \sum_j e^{s_j^k}} \tag{10}$$

The sum of the attention scores for all tokens within each batch can reflect the amount of useful information contained in that batch for the current query. Based on this, we calculate the weighted sum of $\hat{\boldsymbol{h}}_{\boldsymbol{r}}^{\boldsymbol{i}}$ to attain the final output of the attention layer.

4.3 Hyperparameter Searching

To efficiently find suitable values of filtering threshold p and batch size B for different LLMs and tasks, we propose a hyperparameter searching strategy as shown in Algorithm 1. By treating q_i as current query and $\mathcal{S}_{1:i-1}$ as demonstrations, the model perplexity [1] (*ppl*) of r_i can reflect the LLMs' capability when demonstration number is $i-1$ (lower *ppl* indicates better performance). Thus, we choose the p that yields the lowest average *ppl* and B that first leads an increasing trend in *ppl* as our hyperparameter choices. We generally set $\boldsymbol{\mathcal{S}_p}$ as $[0, 0.1, 0.2, 0.3, 0.4]$ and run each setting 5 times to stabilize the results, resulting in a total of 25 inference overhead for hyperparameter searching, which is relatively low compared with the thousands of evaluation samples.

[1]We don't use accuracy because the accuracy obtained under teacher forcing will overestimate the model performance.

Algorithm 1 Hyperparameter Searching.

Require: Candidate filtering threshold set $\boldsymbol{\mathcal{S}_p}$, LLM $\boldsymbol{\mathcal{M}}$
Demonstration set $\boldsymbol{\mathcal{S}_{demos}}$, Demonstration number N
Ensure: Suitable filtering threshold p and batch size B
$D(p, i) \leftarrow 0$ for $p \in \mathcal{S}_p, i \in [0, N-1]$
for $p \in \boldsymbol{\mathcal{S}_p}$ **do**:
  for $i \leftarrow 1, 5$ **do**:
    $\boldsymbol{\mathcal{S}_{1:N}} \leftarrow \texttt{RandomSelect}(\boldsymbol{\mathcal{S}_{demos}}, N)$
    # calculate average *ppl* of responses in $\mathcal{S}_{1:N}$
    $\boldsymbol{ppl_{1:N}} \leftarrow \boldsymbol{\mathcal{M}}(\texttt{ICLFormat}(\boldsymbol{\mathcal{S}_{1:N}}))$
    $D(p, j-1) \leftarrow D(p, j-1) + \boldsymbol{ppl_j}$ for $j \in [1, N]$
  end for
  $D(p, i) \leftarrow D(p, i) + D(p, i+1)$ for $i \in [0, N-2]$
  $\bar{D}(p, i) \leftarrow D(p, i) - D(p, i-2)$ for $i \in [2, N-2]$
end for
$p \leftarrow \texttt{argmin}(p|\texttt{sum}(D(p)))$
$B \leftarrow \texttt{argmin}(i|\bar{D}(p, i) > 0)$

5 Experiments

Centered around FocusICL, we will empirically demonstrate its performance on different LLMs and tasks in §5.2, verify whether it can help LLMs scale well with demonstration number in §5.3, and delve into its working mechanism in §5.4. We also investigate the choice of hyperparameters in Appendix §A.1.

5.1 Experimental Settings

Benchmarks We conduct experiments on the following benchmarks:

- **CSQA** (Talmor et al., 2019) is a high-quality benchmark for commonsense reasoning task.
- **PIQA** (Bisk et al., 2020) concentrates on testing physical commonsense answering ability.
- **CountA** is our proposed benchmark to avoid the impact of data contamination (Jiang et al., 2024), making experimental results more comprehensive and reliable. It requires the model to count the number of character 'A' in the five candidates.
- **ARC** (Clark et al., 2018) includes questions that require extensive knowledge and reasoning to answer.
- **GSM8K** (Cobbe et al., 2021) serves as a testbed for evaluating multi-step mathematical reasoning ability.

We evaluate the LLMs on the test set of the datasets above and use the training set as the demonstration candidate set $\boldsymbol{\mathcal{S}_{demos}}$.

Baselines

- **ICL.** We use a unified ICL (Brown et al., 2020) input format for all the methods for fair comparisons, as shown in Appendix §C.

Method	CSQA	PIQA	CountA	ARC	GSM8K	Avg.
ICL	47.58	57.42	79.04	62.43	9.93	51.28
EarlyStop	47.89	57.44	81.28	62.43	11.14	52.04
StructICL	50.25	59.02	86.77	64.05	11.25	54.27
Triviality	48.97	58.65	85.68	63.13	11.00	53.49
FocusICL	**50.70**	**60.83**	**91.94**	**64.55**	**12.28**	**56.06**

Table 1: Accuracy (%) of LONGCHAT-7B-V1.5-32K with compared methods across benchmarks.

Method	CSQA	PIQA	CountA	ARC	GSM8K	Avg.
ICL	60.72	60.09	82.20	77.11	16.30	59.23
EarlyStop	61.36	60.20	82.20	78.14	17.44	59.87
StructICL	61.44	61.81	84.78	78.05	17.12	60.64
Triviality	61.51	61.03	84.43	77.78	17.36	60.42
FocusICL	**62.57**	**67.88**	**85.13**	**78.51**	**17.74**	**62.37**

Table 2: Accuracy (%) of VICUNA-7B-V1.5-16K with compared methods across benchmarks.

- **EarlyStop.** Zhao et al. (2023) proposes to pick the optimal demonstration number according to the performance on a validation set.
- **StructICL.** Hao et al. (2022) share a similar idea with us of dividing demonstrations into batches. Differently, their designs focus on extending available context length.

Details We conduct experiments with three widely used long-context LLMs: LONGCHAT-7B-V1.5-32K (Dacheng Li* and Zhang, 2023), VICUNA-7B-V1.5-16K (Zheng et al., 2023) and LLAMA-3-8B-INSTRUCT (AI@Meta, 2024). We choose the maximum available number of demonstrations for evaluation based on the 40 GB memory of the A100 GPU (Table 7). The hyper parameter searching results are listed in Table 8. We use random sampling decoding strategy (T=0.1) and report the outcomes averaged over 5 runs (randomly selecting demonstrations) for credible results.

5.2 Main Results

Our main experimental results are presented in Tables 1, 2, and 3. The compared methods exhibit similar performance trends across different LLMs.

Baselines Under most settings, EarlyStop outperforms ICL, consistent with the observations of Agarwal et al. (2024) and Zhao et al. (2023) that more demonstrations does not necessarily lead to better performance. Compared to EarlyStop which avoids the negative impact of attention dispersion by not introducing more demonstrations, StructICL leverages all the given demonstrations through structured input to achieve slightly better performance.

Method	CSQA	PIQA	CountA	ARC	GSM8K	Avg.
ICL	74.90	75.86	98.10	90.00	66.64	81.10
EarlyStop	75.54	77.09	98.10	90.47	71.21	82.48
StructICL	75.12	77.05	98.16	90.70	69.43	82.09
Triviality	75.25	76.38	98.22	90.40	68.03	81.56
FocusICL	**76.00**	**78.29**	**98.34**	**91.02**	**71.89**	**83.11**

Table 3: Accuracy (%) of LLAMA-3-8B-INSTRUCT with compared methods across benchmarks.

Ours However, due to the lack of insights into the reasons behind performance degradation of ICL with more demonstrations, the baselines fail to maintain the model attention on critical input parts while fully leveraging all demonstrations. In contrast, by introducing triviality filtering operation and hierarchical attention mechanism to achieve the above vision, FocusICL outperforms the compared baselines, achieving an average of 5.2% (3.31 points) performance improvement over ICL across three LLMs. This validates the effectiveness and generalizability of FocusICL.

Ablations We also report the performance of only performing triviality filtering operation as an ablation study. The results show that FocusICL benefits 1.29 points improvement from the triviality filtering operation and 2.02 points improvement from the hierarchical attention mechanism.

Efficiency By performing hierarchical attention mechanism, demonstrations between different batches does not need direct interactions, which can save a significant amount of inference overhead. Assuming each demonstration has an average of L tokens, the overhead of attention operation between N demonstrations for ICL is:

$$Cost_{\text{ICL}} = N^2 L^2 \times \Delta \tag{11}$$

where Δ denotes a computational cost unit. The overhead for FocusICL with batch size as B is:

$$\begin{aligned} Cost_{\text{FocusICL}} &= \frac{N}{B}(BL)^2 \times \Delta \\ &= NBL^2 \times \Delta \end{aligned} \tag{12}$$

Therefore, the overhead ratio of FocusICL to ICL in encoding demonstrations is $B : N$ (N is generally several times larger than B), while the overhead in other aspects is roughly the same. This demonstrates the efficiency of FocusICL.

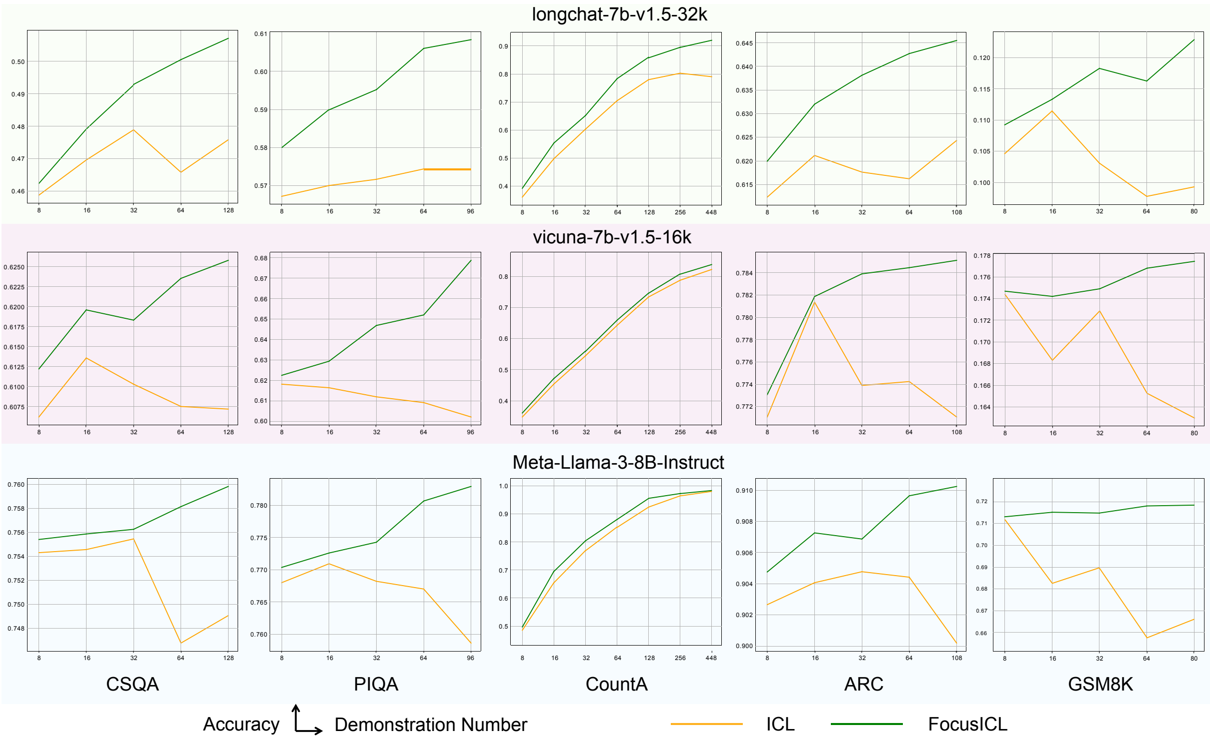


Figure 5: FOCUSICL helps different LLMs scale well with many-shot demonstrations compared with ICL.

5.3 Scaling with More Demonstrations

The recent significant advancements in LLMs mainly stem from scaling up in dimensions of model size and training data size. However, given the limitations of computation resource and data production speed, we are in eager need of exploring other potential scaling dimensions to continuously enhance the performance of LLMs. As shown in Figure 5, the demonstration number is not a stable scaling dimension when applying ICL, as the performance can sometimes exhibit an inverse-scaling phenomenon with more demonstrations. In contrast, FOCUSICL enables LLMs to become stable many-shot learners by directing their attention to important contents, thereby achieving good scalability in the dimension of demonstration number.

It should be noted that we find the advantage of FOCUSICL over ICL continues to grow as the number of demonstrations increases. This means that if we have more resources to conduct experiments with more demonstrations, the advantage of FOCUSICL over ICL can be larger.

5.4 Working Mechanism of FOCUSICL

To gain a deeper understanding of the working mechanism of FOCUSICL, we explore it from aspects of attention and hidden state distributions, following the experimental settings in §3.3.

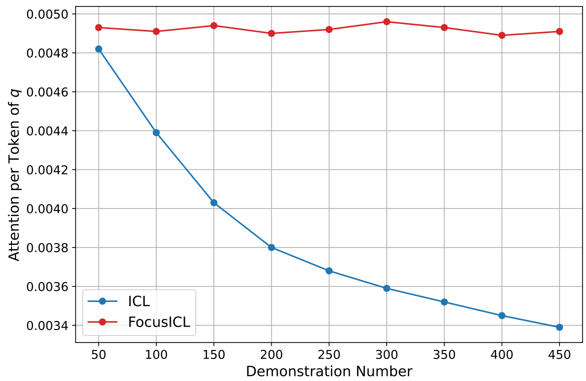


Figure 6: Average model attention towards token of q with varying demonstration numbers.

Attention Distribution The primary purpose of FOCUSICL is to prevent the model attention from being scattered by the increased demonstrations, thereby ensuring a proper understanding of key contents. Therefore, we observe the attention weights allocated by the model towards the query as the number of demonstrations increases. As shown in Figure 6, by ignoring unimportant parts of the demonstrations and introducing the hierarchical attention mechanism, FOCUSICL consistently maintains sufficient attention towards the query.

Hidden States Distribution We further investigate the distribution of the hidden states of the last input token at the penultimate model layer through

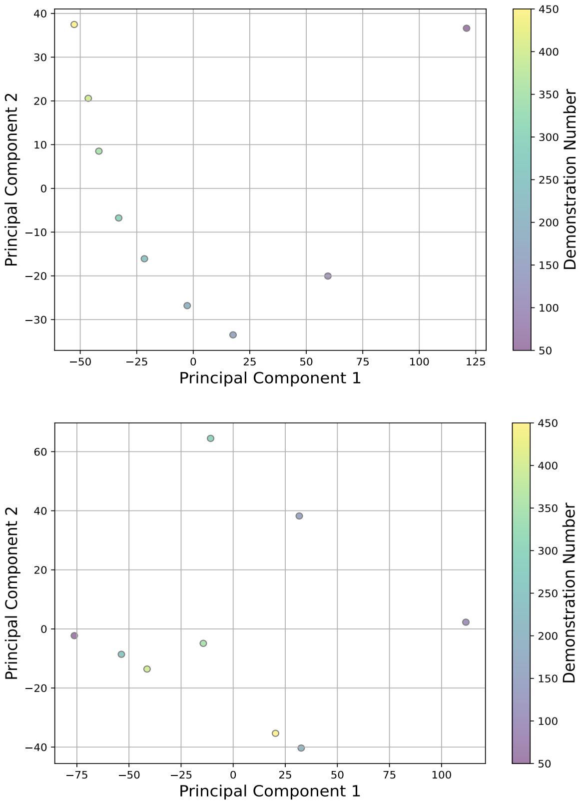


Figure 7: The PCA distribution results of the hidden states of the last input token from the penultimate layer of ICL (above) and FOCUSICL (below) with varying numbers of demonstrations.

Principal Component Analysis (PCA). Intuitively, the distribution of the hidden states of the last token mainly depends on the current problem to be solved and should be independent of the demonstration number. However, as shown in Figure 7, we find that the hidden states of ICL change systematically with an increasing number of demonstrations, whereas FOCUSICL does not exhibit such behavior. We think that the systematic decline in attention towards the query in ICL with an increasing number of demonstrations continuously affects the hidden states during response generation, thereby impacting the quality of the generated response. In contrast, FOCUSICL avoids this issue by maintaining sufficient attention to the query as shown above, ultimately benefiting from more demonstrations.

5.5 Further Discussion

Based on our existing insights and experimental results, we attempt to understand the divergent phenomena of ICL observed in previous studies where more demonstrations sometimes lead to better performance, while sometimes the opposite occurs. We think the main reason leading to the above phenomena comes from the double-edged sword effect of learning from more demonstrations: on the one hand, they can help the model better understand the task and increase the likelihood of finding useful knowledge; on the other hand, they might also distract the model, leading to insufficient attention and understanding of current query. We consider that two aspects can influence the balance between the two effects:

Weak models require more demonstrations to understand the task. As shown in Figure 5, we observe that the optimal number of demonstrations for LONGCHAT-7B-V1.5-32K is greater compared to the other two models across most benchmarks. Considering that its performance is also the worst, we believe the reason for the aforementioned situation is that weaker models require more demonstrations to help them better understand the task.

More demonstrations are needed when they have a closer relationship. We also notice that the LLMs are more demonstration-hungry on CountA compared to other benchmarks as shown in Figure 5. We analyze that the correlation between samples in other benchmarks is relatively weak, and even a single demonstration is sufficient to clarify the task format. In contrast, the demonstrations in CountA are closely related, collectively determining what the task definition is. In this scenario, LLMs cannot discern the complete task information if only given a few demonstrations. To sum up, when the samples are closely related, the model needs more demonstrations to analyze the correlations among them, so as to better understand and complete the task.

6 Conclusions

Noticing that the performance of LLMs under many-shot ICL does not consistently improve with more demonstrations, we analyze and validate the underlying reason as follows: more demonstrations can disperse the model attention to critical contents, resulting in an insufficient understanding of the query. Inspired by how humans learn from examples, we propose a training-free method FOCUSICL, which conducts triviality filtering at token-level and hierarchical attention at demonstration-level to rationally allocate model attention in each layer. Comprehensive experiments indicate that focused LLMs are stable many-shot learners, making demonstration number a possible scaling dimension for LLM-based AGI.

Limitations

From an objective perspective , we think there are two main limitations of this paper:

1. Although we have extended the demonstration number to nearly or even beyond 100, due to computational resource limitations, we are unable to conduct experiments with larger demonstration numbers. We will further verify the applicability of FocusICL with larger demonstration numbers in the future.

2. This work primarily discusses LLMs that apply the standard transformer decoder architecture. We look forward to further exploring the scaling performance with the demonstration number and the applicability of FocusICL on other variants of LLMs, such as the encoder-decoder architecture and sliding window attention, in the future.

Ethics Statement

All of the datasets used in this study were publicly available, and no annotators were employed for our data collection. We confirm that the datasets we used did not contain any harmful content and was consistent with their intended use (research). We have cited the datasets and relevant works used in this study.

References

Rishabh Agarwal, Avi Singh, Lei M. Zhang, Bernd Bohnet, Stephanie C. Y. Chan, Ankesh Anand, Zaheer Abbas, Azade Nova, John D. Co-Reyes, Eric Chu, Feryal M. P. Behbahani, Aleksandra Faust, and Hugo Larochelle. 2024. Many-shot in-context learning. *CoRR*, abs/2404.11018.

AI@Meta. 2024. Llama 3 model card.

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.

Amanda Bertsch, Maor Ivgi, Uri Alon, Jonathan Berant, Matthew R Gormley, and Graham Neubig. 2024. In-context learning with long-context models: An in-depth exploration. *arXiv preprint arXiv:2405.00200*.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: reasoning about physical commonsense in natural language. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7432–7439. AAAI Press.

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*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the AI2 reasoning challenge. *CoRR*, abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168.

Anze Xie Ying Sheng Lianmin Zheng Joseph E. Gonzalez Ion Stoica Xuezhe Ma Dacheng Li*, Rulin Shao* and Hao Zhang. 2023. How long can open-source llms truly promise on context length?

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.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey for in-context learning. *CoRR*, abs/2301.00234.

Hanyu Duan, Yixuan Tang, Yi Yang, Ahmed Abbasi, and Kar Yan Tam. 2023. Exploring the relationship between in-context learning and instruction tuning. *CoRR*, abs/2311.10367.

Yaru Hao, Yutao Sun, Li Dong, Zhixiong Han, Yuxian Gu, and Furu Wei. 2022. Structured prompting: Scaling in-context learning to 1, 000 examples. *CoRR*, abs/2212.06713.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Danny Hernandez, Jared Kaplan, Tom Henighan, and Sam McCandlish. 2021. Scaling laws for transfer. *CoRR*, abs/2102.01293.

Kazuki Irie, Róbert Csordás, and Jürgen Schmidhuber. 2022. The dual form of neural networks revisited: Connecting test time predictions to training patterns via spotlights of attention. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 9639–9659. PMLR.

Minhao Jiang, Ken Ziyu Liu, Ming Zhong, Rylan Schaeffer, Siru Ouyang, Jiawei Han, and Sanmi Koyejo. 2024. Investigating data contamination for pre-training language models. *CoRR*, abs/2401.06059.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 8086–8098. Association for Computational Linguistics.

Ian R. McKenzie, Alexander Lyzhov, Michael Pieler, Alicia Parrish, Aaron Mueller, Ameya Prabhu, Euan McLean, Aaron Kirtland, Alexis Ross, Alisa Liu, Andrew Gritsevskiy, Daniel Wurgaft, Derik Kauffman, Gabriel Recchia, Jiacheng Liu, Joe Cavanagh, Max Weiss, Sicong Huang, The Floating Droid, Tom Tseng, Tomasz Korbak, Xudong Shen, Yuhui Zhang, Zhengping Zhou, Najoung Kim, Samuel R. Bowman, and Ethan Perez. 2023. Inverse scaling: When bigger isn't better. *CoRR*, abs/2306.09479.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy P. Lillicrap, Jean-Baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, Ioannis Antonoglou, Rohan Anil, Sebastian Borgeaud, Andrew M. Dai, Katie Millican, Ethan Dyer, Mia Glaese, Thibault Sottiaux, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, James Molloy, Jilin Chen, Michael Isard, Paul Barham, Tom Hennigan, Ross McIlroy, Melvin Johnson, Johan Schalkwyk, Eli Collins, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, Clemens Meyer, Gregory Thornton, Zhen Yang, Henryk Michalewski, Zaheer Abbas, Nathan Schucher, Ankesh Anand, Richard Ives, James Keeling, Karel Lenc, Salem Haykal, Siamak Shakeri, Pranav Shyam, Aakanksha Chowdhery, Roman Ring, Stephen Spencer, Eren Sezener, and et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *CoRR*, abs/2403.05530.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4149–4158. Association for Computational Linguistics.

Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. 2023. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 35151–35174. PMLR.

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.

Fei Zhao, Taotian Pang, Zhen Wu, Zheng Ma, Shujian Huang, and Xinyu Dai. 2023. Dynamic demonstrations controller for in-context learning. *CoRR*, abs/2310.00385.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. 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*.

A Additional Experimental Results

A.1 Hyperparameters

To investigate the influence of hyperparameters, we report the results of LONGCHAT-7B-V1.5-32K on GSM8K benchmark with varying hyperparameter settings.

Filtering Threshold As shown in Table 4, with the increase of filtering threshold p, the model's performance first improves and then declines. This is because, when p is relatively small, the model benefits from ignoring unimportant content and focusing its attention on more beneficial parts. However, when p becomes larger, the model might overlook potentially useful information in the demonstrations, leading to a decrease in performance.

Batch Size As shown in Table 5, a similar inverted U-shaped curve phenomenon occurs when scaling the batch size while maintaining the overall demonstration number fixed. As the batch size decreases from 80, the model attention to the query continues to increase, which can lead to a certain improvement in model performance. However, when the batch size is too small, the model may fail to fully understand the task definition due to excessive lack of interaction between demonstrations, consistent with the findings of Bertsch et al. (2024).

Luckily, through our proposed hyperparameter searching strategy, we can efficiently attain suitable hyperparameters for the given tasks and LLMs.

A.2 Inverse-scaling Phenomena with Gemini

Due to the limitations of computational resources and the unavailability of closed-source models, our experiments are primarily conducted on 7-8B open source LLMs. However, by utilizing APIs, we additionally explore the performance changes of more powerful models as the number of demonstrations increased, further validating the generalizability of the argument that LLMs are not stable many-shot learners. We choose to experiment with GEMINI 1.5 PRO for its long available context window (1M tokens). We test GEMINI 1.5 PRO on MATH benchmark (Hendrycks et al., 2021), which contains 7 subsets with 5 difficulty levels that can thoroughly evaluating the math reasoning abilities of LLMs. We use greedy searching decoding strategy with and report the outcomes averaged over 5 runs for credible results. As shown in Figure 8, obvious inverse-scaling phenomenon appears in 5 out of 7 subsets, with Precalculus and Intermediate Algebra as exceptions. This validates the generalizability of the argument that LLMs are not stable many-shot learners. Meanwhile, we observe that across different difficulty levels, GEMINI 1.5 PRO presents similar performance changing trends. Figure ?? clearly shows such phenomenon. This indicates that the task difficulty does not affects the optimal demonstration number of certain task.

Filtering Threshold	0.0	0.1	0.2	0.3	0.4
FOCUSICL	11.90	12.28	12.03	12.05	11.88

Table 4: Accuracy (%) of LONGCHAT-7B-V1.5-32K when applying FOCUSICL with varying filtering threshold and batch size as 8.

Batch Size	2	4	8	16	80
FOCUSICL	10.46	10.99	12.28	11.45	11.00

Table 5: Accuracy (%) of LONGCHAT-7B-V1.5-32K when applying FOCUSICL with varying batch sizes and filtering threshold as 0.1. It should be noted that the overall demonstration number is fixed as 80.

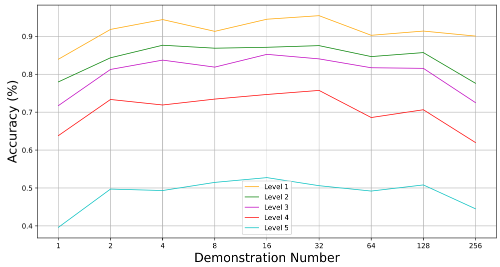
Method	CSQA	PIQA	CountA	ARC	GSM8K
N	128	96	448	108	80

Table 6: The total demonstration number N of different benchmarks in our experiments.

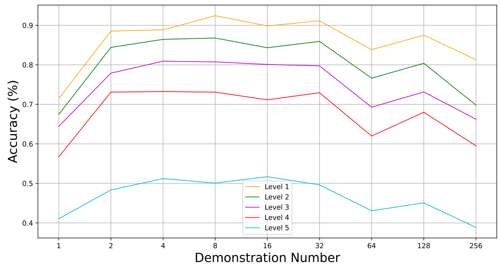
B Derivation Details

The derivation details of Equation (5) are as follows:

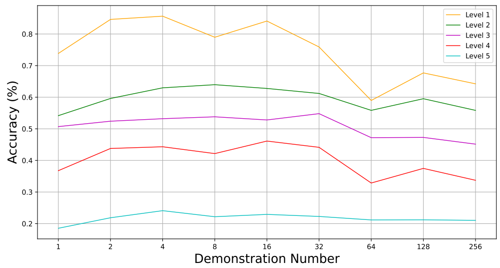
$$
\begin{aligned}
&\text{output} \\
=&\,\mathrm{Att}(\boldsymbol{h_r}\boldsymbol{W}_q, \mathrm{Cat}[\boldsymbol{D}_k;\boldsymbol{Q}_k], \mathrm{Cat}[\boldsymbol{D}_v;\boldsymbol{Q}_v]) \\
=&\,\mathrm{softmax}(\boldsymbol{h_r}\boldsymbol{W}_q\,\mathrm{Cat}[\boldsymbol{D}_k;\boldsymbol{Q}_k]^\top)\begin{bmatrix}\boldsymbol{D}_v\\ \boldsymbol{Q}_v\end{bmatrix} \\
=&\,\frac{\sum_j \exp\left(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{Q}_k^\top\right)_j}{\sum_i \exp\left(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{D}_k^\top\right)_i + \sum_j \exp\left(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{Q}_k^\top\right)_j} \\
&\times \mathrm{softmax}(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{Q}_k^\top)\boldsymbol{Q}_v \\
&+\frac{\sum_i \exp\left(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{D}_k^\top\right)_i}{\sum_i \exp\left(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{D}_k^\top\right)_i + \sum_j \exp\left(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{Q}_k^\top\right)_j} \\
&\times \mathrm{softmax}(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{D}_k^\top)\boldsymbol{D}_v \\
=&\,(1-\lambda(\boldsymbol{h_r}))\,\mathrm{softmax}(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{Q}_k^\top)\boldsymbol{Q}_v \\
&+\lambda(\boldsymbol{h_r})\,\mathrm{softmax}(\boldsymbol{h_r}\boldsymbol{W}_q\boldsymbol{D}_k^\top)\boldsymbol{D}_v \\
=&\,(1-\lambda(\boldsymbol{h_r}))\underbrace{\mathrm{Att}\left(\boldsymbol{h_r}\boldsymbol{W}_q,\boldsymbol{Q}_k,\boldsymbol{Q}_v\right)}_{\text{outcome from } \boldsymbol{q}} \\
&+\lambda(\boldsymbol{h_r})\underbrace{\mathrm{Att}\left(\boldsymbol{h_r}\boldsymbol{W}_q,\boldsymbol{D}_k,\boldsymbol{D}_v\right)}_{\text{outcome from } \boldsymbol{demos}},
\end{aligned}
\tag{13}
$$



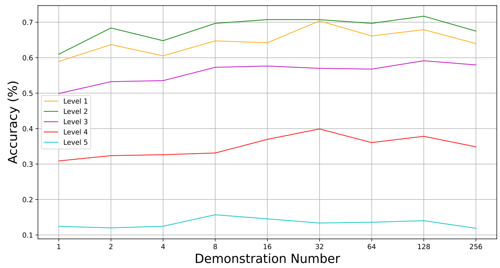
(a) Algebra



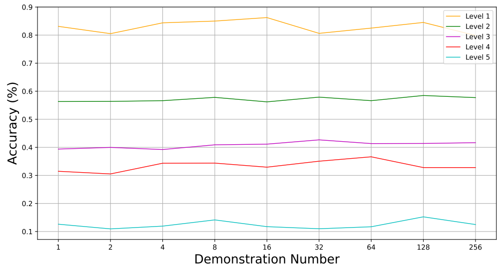
(b) Prealgebra



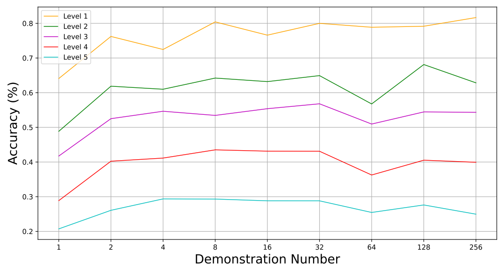
(c) Counting and Probability



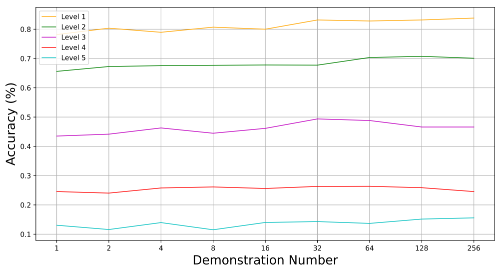
(d) Geometry



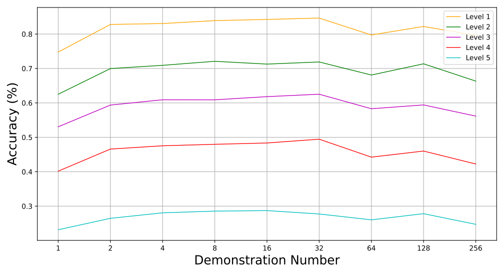
(e) Intermediate Algebra



(f) Number Theory



(g) Precalculus



(h) Average

Figure 8: Performance of Gemini on different subset of MATH with varying demonstration numbers.

where:

$$\lambda(\boldsymbol{h_r}) = \frac{\sum_i \exp\left(\boldsymbol{h_r} \boldsymbol{W}_q \boldsymbol{D}_k^\top\right)_i}{\sum_i \exp\left(\boldsymbol{h_r} \boldsymbol{W}_q \boldsymbol{D}_k^\top\right)_i + \sum_j \exp\left(\boldsymbol{h_r} \boldsymbol{W}_q \boldsymbol{Q}_k^\top\right)_j} \tag{14}$$

C Prompt Template

The following is a template ICL input format when demonstration number is 2.

Human: I'm getting warm because I increased the thermostat in my bedroom. What might I be doing soon? Answer

Method	**CSQA**	**PIQA**	**CountA**	**ARC**	**GSM8K**
Training size	9741	16113	3000	2241	7473
Testing size	1221	1838	1000	567	1319

Table 7: Benchmark Statistics.

Model	LONGCHAT-7B -V1.5-32K		VICUNA-7B -V1.5-16K		LLAMA-3-8B -INSTRUCT	
Params	p	B	p	B	p	B
CSQA	0.1	32	0.2	16	0.4	32
PIQA	0.1	32	0.1	8	0.4	2
CountA	0.4	112	0.4	224	0.4	112
ARC	0.4	16	0.4	0.1	0.4	12
GSM8K	0.1	8	0.1	8	0.4	8

Table 8: The results of hyperparameter searching strategy across varing tasks and LLMs.

Choices: (a) feeling comfortable (b) overheat (c) increase of temperature (d) pleasure (e) starting fire

Assistant: A

Human: Where might I hear and see information on current events? Answer Choices: (a) internet (b) television (c) newspaper (d) book (e) radio

Assistant: B

Human: If somebody buys something and gives it to me as a free gift, what is the cost status of the gift? Answer Choices: (a) deadly (b) imprisoned (c) paid for (d) expensive (e) in prison

Assistant: