Multi-Layer Attention is the Amplifier of Demonstration Effectiveness

Anonymous ACL submission

Abstract

Many studies show that not all demonstrations help in-context learning (ICL), limiting performance. Therefore, in this paper, we analyze why demonstrations become ineffective using gradient flow. By setting the gradient flow to zero, we reveal two cases of ineffectiveness: the model has either already learned the information or it is irrelevant to the query. We also prove that in the multi-layer attention model, effectiveness disparities amplify with depth, directing attention toward effective demonstrations. Building on the above discussion, we propose GRADS, which selects demonstrations via gradient-flow signals and explicitly accounts for already assimilated information. We validate our derivation and GRADS on four prominent LLMs across five mainstream datasets. The experiment confirms that the disparity in demonstration effectiveness is magnified as the model layer increases, substantiating our derivations. Moreover, GRADS achieves a relative improvement of 1.3% on average over the strongest baselines, achieving new SOTA results in the demonstration selection.

1 Introduction

In-Context Learning (ICL) is an effective method for enhancing the performance of Large Language Models (LLMs) (Brown et al., 2020; Dong et al., 2024). By providing demonstrations relevant to the user query in the input, ICL guides the reasoning of LLMs, thereby improving inference performance. Recent years have witnessed many efforts in investigating the internal mechanisms of ICL for guiding the design of ICL methods (Zhou et al., 2024). For example, recent research works (Zhang et al., 2024; Mahankali et al., 2024; Smart et al., 2025) discuss the convergence and convergence speed of ICL, while some other works (Olsson et al., 2022; Li et al., 2024b; Chen et al., 2024) study the function of each part and module of the Transformer (Vaswani et al., 2017) during ICL.

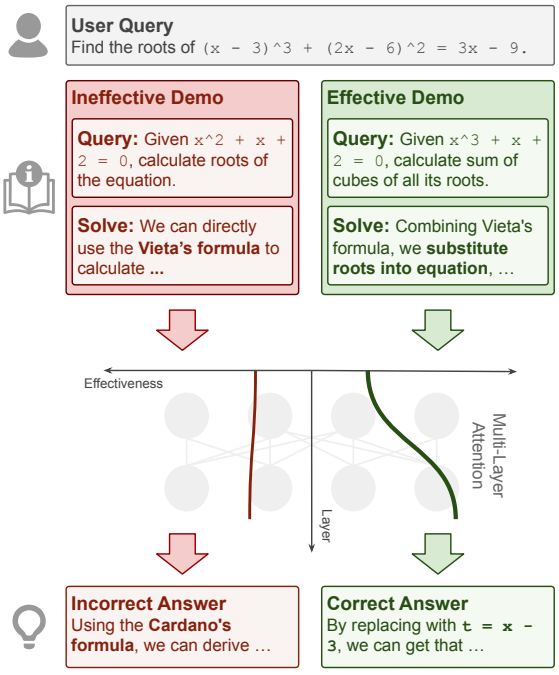

Figure 1: The gradient flow comparison of effective (green) and ineffective demonstrations (red). For the ineffective demonstration, with the increase of model layers, the effectiveness remains at a low level. About the effective ones, the effectiveness is significantly amplified as the layer increases.

As the performance of LLMs continues to improve, a phenomenon emerges that there exist ineffective demonstrations to the given query, which cannot enhance reasoning performance during ICL (DeepSeek-AI, 2025; Wang et al., 2025c). However, existing research on the mechanisms of ICL is predominantly based on the assumption that the given demonstrations are not ineffective (Liu et al., 2025; Cho et al., 2025), which limits the exploration of how to enhance the performance of ICL. Therefore, in this paper, we aim to answer the following research questions (RQs): *1) What makes the demonstration ineffective? 2) How does the effectiveness of the demonstration affect the ICL performance?* and *3) How to select effective demonstrations to enhance ICL performance?*

Following Wang et al. (2023); Liu and Deng (2025), we employ the gradient flow for our investigation: the greater the gradient flow from the demonstration to the generated answer, the demonstration is more effective. We first study the factors that determine the magnitude of the gradient flow in a single-layer linear self-attention model (LSA), thereby providing the reasons for an ineffective demonstration: the information in the demonstration has already been learned by LLMs, or the demonstration is irrelevant to the user query (RQ1). We then prove that in a multi-layer LSA, **the ratio of the accumulated gradient flow among demonstrations is amplified along the increase of the number of layers** (RQ2), as shown in Figure 1.

Considering that existing demonstration selection methods mainly focus on the relevance between the demonstration and the query, which could select ineffective demonstrations (Wei et al., 2024; Chen et al., 2023a; Wang et al., 2025b). Therefore, based on the above discussion, we propose GRADS, which selects the demonstration that maximizes the gradient flow with the given query (RQ3). Measure whether LLMs utilize the information in a given demonstration through the gradient flow to ensure that the demonstrations provided are effective in reasoning, enhancing performance.

To validate our conclusions and the effectiveness of GRADS, we conduct experiments on five mainstream datasets and four mainstream LLMs. Experimental analysis shows that the ratio of the gradient flow between effective and ineffective demonstrations increases with the model layers, proving that the multi-layer Transformer indeed acts as an amplifier of demonstration effectiveness. Furthermore, the experimental results of GRADS indicate that, compared to the best demonstration selection baselines, GRADS achieves a relative improvement of 1.3% on average, achieving new SOTA results on the demonstration selection.

In summary, our contributions include:

- We prove that a demonstration is ineffective when the information has already been learned by the LLMs or is irrelevant to the given user query.
- We theoretically and empirically analyze that the multi-layer structure amplifies the ratio of the demonstration effectiveness.
- We propose GRADS, a gradient-based demonstration selection method that achieves new SOTA results on the demonstration selection.

2 Analysis of Demonstration Effectiveness

In this section, we discuss *why* and *how* LLMs ignore ineffective demonstrations in ICL from a gradient flow perspective. We first provide definitions for necessary mathematical notations and concepts (§2.1). Then, we discuss the gradient flow in a single-layer linear self-attention network and why LLMs distinguish ineffective demonstrations (§2.2). Subsequently, we discuss the gradient flow in a multi-layer linear self-attention network and how LLMs ignore ineffective demonstrations (§2.3). The main findings of our analysis are summarized in Table 1. All proofs in this section are provided in Appendix A, and we follow Wang et al. (2023) to calculate the gradient flow.

2.1 Preliminary

Without loss of generality, we primarily focus on the 1-shot setting for mechanism analysis. Following Zhang et al. (2024), we denote a demonstration as $d = \begin{pmatrix} d_x & d_y \end{pmatrix}$, where $d_x, d_y \in \mathbb{R}^e$ represents the input and output embedding vector of the demonstration, respectively. Here, e is the embedding dimension. We denote a user query as $q = \begin{pmatrix} q_x & q_y \end{pmatrix}$, where $q_x \in \mathbb{R}^e$ is the query input embedding vector, and we set $q_y = 0$ to indicate that the corresponding answer to the query is not provided in the input. We denote the complete network input as $E = \begin{pmatrix} d & q \end{pmatrix}$. In this paper, we use $\|M\|$ to denote the Frobenius norm (Wu et al., 2022) of a given matrix M.

Following previous work (Zhang et al., 2024), we define the linear self-attention network (LSA):

$$f_{LSA}(E;\theta) = E + W^{PV}E \cdot \frac{E^\top W^{KQ} E}{\rho} \quad (1)$$

$W^{PV} \in \mathbb{R}^{e\times e}$ is the combined projection and value matrix of Attention, and $W^{KQ} \in \mathbb{R}^{e\times e}$ is the combined key and query matrix, where the specific definitions are consistent with Zhang et al. (2024). ρ is the normalization coefficient, where we set $\rho = 1$ in this paper following Zhang et al. (2024). We denote all network parameters as $\theta = \{W^{PV}, W^{KQ}, \rho\}$. It can be observed that Equation 1 is the result of replacing the activation function in a single-layer attention network of a Transformer with a linear function. Specifically, we denote $\hat{q_y}(E;\theta)$ as the predicted answer of the given E and θ, abbreviated as $\hat{q_y}$, which is the vector corresponding to the last column of the output from the LSA model $f_{LSA}(E;\theta)$.

RQ	Finding	Evidence
RQ1: What makes the demonstration ineffective?	A demonstration is ineffective if the information it contains has already been learned by LLMs or is irrelevant to the user query.	Equation 2
RQ2: How does the effectiveness of the demonstration affect the ICL performance?	With deeper layers, the ratio of the accumulated gradient flow between effective and ineffective demonstrations widens.	Theorem 1

Table 1: The main research questions (RQs), findings, and corresponding evidence of our analysis.

Following previous work (Wang et al., 2023), for a multivariate function f and one of its independent variables x, we define the magnitude of the gradient flow of f with respect to x as the partial derivative of f with respect to x, which is $\nabla_x f = \frac{\partial f}{\partial x}$. The magnitude of the gradient flow measures the influence of the change in x on f. Specifically, in the context of ICL, the gradient flow of the output $\hat{q_y}$ with respect to an input demonstration d reflects the contribution of that demonstration to the answer. Consequently, it can indicate how much information from the demonstration is utilized during generating the answer.

2.2 Single-Layer Linear Self-Attention

We first analyze the gradient flow in a single-layer network. It can be shown that the contribution of the input demonstration to the gradient flow for answer generation is:

$$\nabla_d \hat{q} = (W^{PV} d)(q^\top W^{KQ})^\top + (d^\top W^{KQ} q) W^{PV} \tag{2}$$

Equation 2 formally defines the gradient flow value of the single layer LSA, which indicates how much information from the demonstration is used for answer generation. We can see that, given a certain query q, the factors determining the magnitude of the equation can be divided into two categories: *(i)* $d^\top W^{KQ} q$: The similarity between the demonstration and the user query. *(ii)* $W^{PV} d$: Whether the model has already learned the information in the demonstration (Petroni et al., 2019; Roberts et al., 2020). Therefore, the determination of whether to use the information from a given demonstration mainly depends on *1) the similarity between the demonstration and the query*, and on *2) whether the information in the demonstration has already been learned by the model*.

Based on the discussion above, as the base of the following discussion, we propose to use the parameters in Equation 2 to formally define the effectiveness of demonstrations with the given query and model as follows:

Definition 1 (Demonstration Effectiveness)**.** *Given a query q and model parameters θ, if two demonstrations d_1 and d_2 satisfy that:*

$$\|W^{PV} d_1\| \geq \|W^{PV} d_2\|$$

$$\|d_1^\top W^{KQ} q\| \geq \|d_2^\top W^{KQ} q\|$$

then we say that d_1 is more effective than d_2 with respect to q and θ, denoted as:

$$d_1 \succcurlyeq_{q;\theta} d_2$$

In Definition 1, we require a demonstration to be more effective than the other if it contains more knowledge that the model has not learned ($\|W^{PV} d_1\| \geq \|W^{PV} d_2\|$), and it is more similar to the query ($\|d_1^\top W^{KQ} q\| \geq \|d_2^\top W^{KQ} q\|$). If only one of these conditions holds, it is difficult to compare the effectiveness of the demonstration because it is not possible to determine whether the knowledge or the similarity to the query has a greater impact on performance. We experimentally evaluate the impact of two conditions in Definition 1 on ICL performance in Figure 2.

2.3 Multi-Layer Linear Self-Attention

Considering that current mainstream LLMs consist of multi-layer modules, in this part, we discuss the gradient flow of the multi-layer linear self-attention network. Let L be the total number of layers in the network. We denote the input to the l-th layer as $E^{(l)} = \begin{pmatrix} d^{(l)} & q^{(l)} \end{pmatrix}$ and its parameters as $\theta^{(l)}$, the corresponding predicted answer of the l-th layer is $\hat{q}^{(l)}$. Since in the multi-layer LSA, the output of the $(l-1)$-th layer is the input of the l-th layer:

$$E^{(l)} = f_{LSA}(E^{(l-1)}; \theta^{(l-1)}) \tag{3}$$

According to the chain rule, we can derive that the gradient flow of the whole model is the product of the gradient flow of each layer:

$$\frac{\partial \hat{q}_y^{(L)}}{\partial d^{(0)}} = \frac{\partial \hat{q}_y^{(L)}}{\partial E^{(L-1)}} \times \frac{\partial E^{(L-1)}}{\partial E^{(L-2)}} \times \cdots \times \frac{\partial E^{(1)}}{\partial d^{(0)}} \tag{4}$$

Due to the complexity of the structure, we only provide a qualitative analysis for the multi-layer model. Based on Equation 2, we know that the magnitude of each term in Equation 4 is positively correlated with the demonstration effectiveness, i.e.:

Lemma 1. *Let an L-layer LSA network be given. For every layer $l = 1, \ldots, L$, assume there exist* strictly increasing *functions*

$$g_l : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}, \qquad h_l : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0},$$

such that for every *demonstration d and query q, the following equalities hold:*

$$\begin{aligned} &\|W^{PV,(l)} d^{(l)}\| \\ &= g_l\big(\|W^{PV,(l-1)} d^{(l-1)}\|\big) \end{aligned} \tag{5}$$

$$\begin{aligned} &\|(d^{(l)})^\top W^{KQ,(l)} q^{(l)}\| \\ &= h_l\big(\|(d^{(l-1)})^\top W^{KQ,(l-1)} q^{(l-1)}\|\big) \end{aligned} \tag{6}$$

If two demonstrations satisfy $d_1 \succcurlyeq_{q;\theta^{(0)}} d_2$ at the input layer, then for every *$l = 1, \ldots, L$, we shave $d_1^{(l)} \succcurlyeq_{q;\theta^{(l)}} d_2^{(l)}$.*

The condition in Lemma 1 assumes that each layer of the model has a consistent effect on its input. Based on Lemma 1, we know that for each layer, a more effective input demonstration results in a larger corresponding gradient flow in all layers. **It is worth noting that for the same demonstration d, Lemma 1 does not guaranty that the gradient flow at layer l is always greater than that at layer $l-1$.** This is because the magnitude of the gradient flow also depends on the model parameters $\theta^{(l)}$ of each layer.

Based on Lemma 1, we can deduce that since the gradient flow is a partial derivative, a larger gradient flow leads to a greater change in the output of subsequent layers. Therefore, as the number of model layers increases, the difference in gradient flow between effective and ineffective demonstrations also becomes larger.

Theorem 1 (Multi-Layer Attention is the Amplifier of Demonstration Effectiveness)**.** *For a given user query q and a model θ, let d_1 and d_2 be two demonstrations with corresponding inputs E_1 and E_2. If the condition of Lemma 1 holds, for any $L \geq l_1 > l_2 \geq 1$, we can conclude that the following inequalities hold:*

$$\frac{\|\nabla_{d^{(0)}} \hat{q_y}^{(l_1)}(E_1;\theta)\|}{\|\nabla_{d^{(0)}} \hat{q_y}^{(l_1)}(E_2;\theta)\|} \geq \frac{\|\nabla_{d^{(0)}} \hat{q_y}^{(l_2)}(E_1;\theta)\|}{\|\nabla_{d^{(0)}} \hat{q_y}^{(l_2)}(E_2;\theta)\|}$$

Theorem 1 shows that as the layers increase, the **ratio of the accumulated gradient flow** of different demonstrations also becomes increasingly large. This indicates that *the multi-layer LSA acts as an amplifier for the effectiveness*.

3 GRADS

In this section, we introduce a demonstration selection method, named GRADS, based on gradient flow, selecting the demonstration with the largest flow to the given query as the selection result. The prompt we used is shown in Appendix B.1. Considering Theorem 1, in actual selection, we use the gradient flow of the last layer as the selection metric. Because the effectiveness differences between demonstrations are sufficiently amplified, it allows a better distinction between effective and ineffective demonstrations. We discuss the efficiency of GRADS in Appendix C.1 and Appendix D.3.

However, performing a full inference pass on the model to obtain the gradient flow for each user query results in low computational efficiency. From Equation 2, it can be observed that in the calculation of the gradient flow, the computations for the demonstration and the user query are relatively independent. Therefore, we can first compute the encoded vectors for the demonstrations and the user query separately, and then use these results to calculate the magnitude of the gradient flow through matrix operations. This approach significantly reduces the computational overhead, thereby enhancing the efficiency of demonstration selection.

Specifically, for a given demonstration pool, we first input each demonstration individually to extract the encoding result of the final layer as $\hat{d}$. Then, for each user query, we also input it to extract the encoding result of the final layer as $\hat{q}$. Subsequently, the computed $\hat{d}$ and $\hat{q}$ are substituted into Equation 2 to obtain $\nabla_{\hat{d}} \hat{q_y}^{(L)}$. Since $\nabla_{\hat{d}} \hat{q_y}^{(L)}$ is a $2 \times e$ matrix, containing the gradient flows for both the input and output parts of the demonstration, and considering that both parts are equally important, we use $\|\nabla_{\hat{d}} \hat{q_y}^{(L)}\|$ as the metric for demonstration selection to balance their importance.

4 Experiment

Our experiments are primarily divided into three parts: *(i)* Introduction of the experimental settings and baselines. *(ii)* Verification of Theorem 1 and its related corollaries. *(iii)* Validation of effectiveness and the impact of different factors on GRADS.

4.1 Experiment Setup

Dataset To thoroughly validate our analytic conclusions and the effectiveness of our proposed method, we conduct experiments on five mainstream datasets that span various tasks and domains, including: *(i)* math: GSM8K (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021); *(ii)* reasoning: ARC-Challenge (Yadav et al., 2019) and MMLU-Pro (Wang et al., 2024b); *(iii)* domain-specific: Amazon Review (Ni et al., 2019) and FinQA (Tao et al., 2024). Detailed descriptions of these datasets are provided in Appendix B.2. We employ Exact Match (EM) as the evaluation metric across all experimental datasets.

Model We validate our discovery and method on four LLMs: Llama-3.1-8B/70B-Insturct (Llama-3.1-8B/70B) (Aaron Grattafiori, 2024), Deepseek-R1-Distill-Llama-8B (Llama-R1-8B) (DeepSeek-AI, 2025), and Qwen3-8B (An Yang, 2025)[1]. Our selection of models encompasses various scales, series, and capabilities for generating long chains of thought (Long-CoT, Llama-R1 and Qwen3) (Chen et al., 2025a), which can fully evaluate the effectiveness of our conclusion.

Baseline We compare GRADS with following demonstration selection baselines: BM25 (Li et al., 2023), MMR (Ye et al., 2023), MoD (Wang et al., 2024a), Influence (Nguyen and Wong, 2023), DICL (Kapuriya et al., 2025), GenICL (Zhang et al., 2025c), TopicK (Kweon et al., 2025), and ICL-Grad (Zhang et al., 2025d). The above covers the most advanced current demonstration selection methods. Detailed introductions and configurations for each baseline are provided in Appendix B.3.

Implementation Detail For the mechanism analysis experiments, we employ a 1-shot setting to align with the setup in our theoretical analysis. For the validation of GRADS, we follow previous works (Wang et al., 2025b) and adopt a 3-shot setting to ensure the comparability of the final performance. Following DeepSeek-AI (2025), we set the maximum generation length to 32768, and for each question, we sample a single answer. Our experiments are performed on a single A100-80G GPU, with the selection and inference for each dataset taking approximately one hour on average. All implementations are based on Transformers (Wolf et al., 2020) and vLLM (Kwon et al., 2023).

[1]We utilize the thinking mode of Qwen3 for a comprehensive evaluation.

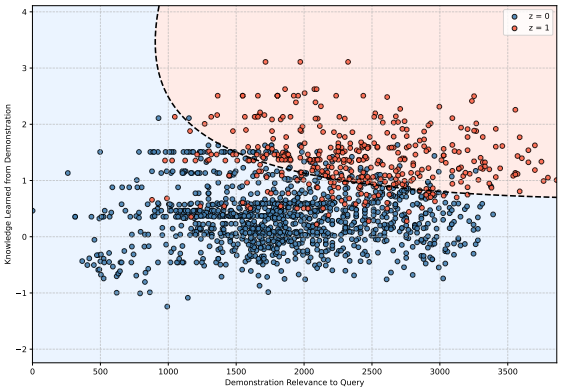

Figure 2: The ICL performance of Llama-3.1-8B across all datasets with different demonstration relevance (X-axis) and unlearned knowledge within the demonstration (Y-axis). Red points denote correct prediction and blue points incorrect predictions. The dashed line represents the decision boundary generated with polynomial logistic regression (Buitinck et al., 2013).

4.2 Experiment of Mechanism Analysis

4.2.1 The Factors Making Demonstrations Ineffective (RQ1)

To verify the condition for low gradient flow in the discussion regarding Equation 2, we record the statistics of the model performance as a function of the changes in demonstration relevance ($d^\top W^{KQ} q$) and learned knowledge ($W^{PV} d$). The experimental results are shown in Figure 2. From the figure, we can observe that: *(i)* As the relevance of the demonstrations and the knowledge learned from them increase, the number of correctly solved data points also increases, which verifies the conclusion from Equation 2 that the performance of ICL is positively correlated with these two factors. *(ii)* ICL only begins to correctly solve the given user queries after the relevance of the demonstrations and the knowledge learned from them surpasses a certain threshold, which indicates that a sufficient amount of effective information relevant to the user query is necessary to enable the model to perform correct reasoning process.

4.2.2 Gradient Flow is Amplified as the Number of Layer Increases (RQ2)

To validate the correctness of Lemma 1 and Theorem 1, we compute the gradient flow in different settings. For a given dataset, we first select the data points that are incorrectly predicted under the 0-shot setting. From this subset, we then separate the data into two groups based on the 1-shot performance: those that are correctly predicted (*effective demonstrations*) and those that remain incorrectly

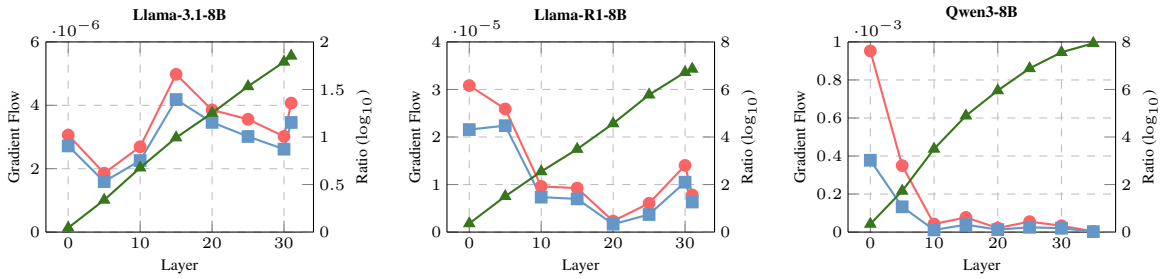

Figure 3: The average gradient flow (left y-axis) and the ratio of the accumulated flow $\frac{\|\nabla_{d(0)} \hat{q_y}^{(l_1)}(E_1;\theta)\|}{\|\nabla_{d(0)} \hat{q_y}^{(l_1)}(E_2;\theta)\|}$ of Theorem 1 (right y-axis) cross different datasets under the i-th layer of each model, where E_1 denotes the input of the effective demonstrations and E_2 denotes the ineffective ones. blue and Red points denote the gradient flow under each layer of the ineffective and effective demonstrations. Green points denote the ratio of the accumulated flow.

predicted (*ineffective demonstrations*). We then compute the average gradient flow for each layer across these two groups. The number of selected effective and ineffective data points on each dataset is detailed in Appendix B.4.

The experimental results are presented in Figure 3. From the figure, we can observe the following: *(i)* Across all settings, the average gradient flow of effective demonstrations is stronger than that of ineffective demonstrations at every layer, which validates the conclusion of Lemma 1. Furthermore, the ratio of gradient flow between effective and ineffective demonstrations increases with the layer depth, which supports Theorem 1. *(ii)* The magnitude of the gradient flow, as well as the accumulated ratio, is significantly more pronounced in the Long-CoT model compared to other models. This suggests that such models are more adept at capturing information from demonstrations and can better distinguish between effective and ineffective demonstrations, showing a higher sensitivity to the useful information contained within the demonstrations. *(iii)* In all models, there are specific layers where the gradient flow exhibits a significant increase. This indicates that the learning from demonstrations is primarily concentrated in these particular layers. While the specific layers differ across models, a strong gradient flow is consistently observed in the final few layers, suggesting that the model pays special attention to the information in the demonstrations when generating the output. *(iv)* The curved boundary shows that either high relevance with non-trivial learning or moderate relevance with substantial learning yields effectiveness during ICL, while high relevance alone with little new knowledge often remains ineffective. Hence, demonstration selection should combine relevance with a learning signal rather than rely on the relevance alone.

4.3 Experiment of GRADS (RQ3)

4.3.1 GRADS is More Effective than Baselines

To validate the effectiveness of GRADS, we conduct a comparative analysis against several baselines, with the experimental results presented in Table 2. The results show that GRADS achieve a relative improvement of 1.3% on average over the best-performing baseline across various models and datasets, achieving new demonstration selection SOTA, which substantiates its efficacy and generalizability. Furthermore, a deeper analysis of the results reveals several key observations:

Baseline In some settings, the performance of other baselines does not exceed that of simpler methods like BM25, and is even inferior to the zero-shot approach in some cases. This suggests that methods based on the similarity between demonstrations and the user query do not guarantee an enhancement in ICL performance due to the fact that the model could have already been exposed to the information present in the demonstrations, and irrelevant information within these demonstrations could consequently mislead the model's reasoning process. In contrast, GRADS ensures that the information in the selected demonstrations is effectively utilized during inference, securing the effectiveness of the selected demonstrations.

Model The most significant performance improvement from GRADS is observed in models without Long-CoT in most settings (e.g., Llama-3.1). This is because such models do not employ Long-CoT, rendering them less capable of effectively leveraging the useful information within the demonstrations. Consequently, their performance is more dependent on the quality of the demonstrations compared to Llama-R1-8B and Qwen3-8B.

Model	Method	GSM8K	MATH	ARC-C	MMLU-Pro	Amazon	FinQA
Llama-3.1-8B	Zero	83.7	47.0	82.0	50.4	63.5	48.1
	BM25 (Li et al., 2023)	84.1	44.6	84.7	54.0	68.4	50.5
	MMR (Ye et al., 2023)	83.9	45.2	85.1	53.6	67.1	50.2
	MoD (Wang et al., 2024a)	84.0	47.0	84.9	53.8	68.7	50.9
	Influence (Nguyen and Wong, 2023)	84.1	47.4	84.6	54.3	68.2	50.8
	DICL (Kapuriya et al., 2025)	84.3	47.4	85.0	54.7	68.9	51.3
	GenICL (Zhang et al., 2025c)	84.5	47.6	85.2	55.0	69.1	51.6
	ICL-Grad (Zhang et al., 2025d)	84.1	47.0	84.8	53.2	67.5	50.4
	TopicK (Kweon et al., 2025)	83.9	47.0	84.1	52.6	66.8	49.8
	GRADS	**85.6**	**48.2**	**86.3**	**56.0**	**70.0**	**52.3**
Llama-3.1-70B	Zero	87.3	63.6	85.2	56.3	69.0	58.3
	BM25 (Li et al., 2023)	86.9	63.2	85.9	57.7	70.8	63.4
	MMR (Ye et al., 2023)	87.2	63.8	86.3	57.4	70.4	63.2
	MoD (Wang et al., 2024a)	87.4	64.2	86.2	57.6	71.0	63.6
	Influence (Nguyen and Wong, 2023)	87.5	64.0	86.6	58.0	71.2	64.0
	DICL (Kapuriya et al., 2025)	87.6	64.0	86.4	57.9	71.1	63.8
	GenICL (Zhang et al., 2025c)	87.8	64.4	86.7	58.3	71.5	64.3
	ICL-Grad (Zhang et al., 2025d)	87.4	64.0	86.0	57.0	70.1	62.0
	TopicK (Kweon et al., 2025)	87.3	63.6	85.7	56.6	69.6	60.8
	GRADS	**88.9**	**65.2**	**87.8**	**59.0**	**72.4**	**65.1**
Llama-R1-8B	Zero	86.0	74.6	83.5	58.2	61.5	60.2
	BM25 (Li et al., 2023)	80.2	74.0	84.0	52.9	65.2	62.9
	MMR (Ye et al., 2023)	85.5	73.0	84.2	58.4	65.9	63.2
	MoD (Wang et al., 2024a)	85.7	74.4	84.0	57.9	65.8	62.6
	Influence (Nguyen and Wong, 2023)	85.8	74.4	84.1	58.6	66.1	63.0
	DICL (Kapuriya et al., 2025)	85.9	74.4	84.3	58.5	66.0	63.4
	GenICL (Zhang et al., 2025c)	86.1	74.6	84.6	58.7	66.2	63.6
	ICL-Grad (Zhang et al., 2025d)	86.0	74.6	83.8	58.3	65.4	62.0
	TopicK (Kweon et al., 2025)	86.0	74.6	83.6	58.2	64.7	61.4
	GRADS	**87.2**	**75.6**	**85.7**	**59.5**	**67.0**	**64.4**
Qwen3-8B	Zero	92.9	76.2	89.2	64.9	62.5	59.0
	BM25 (Li et al., 2023)	92.2	77.2	91.0	65.3	71.5	65.5
	MMR (Ye et al., 2023)	92.5	77.6	90.7	66.5	74.0	66.6
	MoD (Wang et al., 2024a)	92.6	78.2	90.8	66.1	72.0	66.0
	Influence (Nguyen and Wong, 2023)	92.8	78.0	91.2	66.6	73.0	66.5
	DICL (Kapuriya et al., 2025)	92.7	78.0	91.1	67.0	73.8	67.1
	GenICL (Zhang et al., 2025c)	93.0	78.4	91.5	67.6	74.1	67.3
	ICL-Grad (Zhang et al., 2025d)	92.9	76.4	90.2	65.2	71.8	65.0
	TopicK (Kweon et al., 2025)	92.9	76.2	89.7	65.0	70.9	64.2
	GRADS	**94.2**	**79.4**	**92.6**	**68.5**	**75.0**	**68.2**

Table 2: The performance of GRADS compared with different baselines. ARC-C denotes ARC-Challenge, Amazon denotes Amazon Review. Zero denotes the zero-shot setting. The best result under each setting is marked in **bold**.

Dataset GRADS yielded more substantial performance gains on the MATH and MMLU-Pro datasets. This is because such two datasets demand a higher degree of specialized knowledge, which could not be contained in LLMs. GRADS, being more effective at selecting demonstrations that contain the requisite knowledge for the model, achieves a more significant performance improvement in these contexts.

4.3.2 The Performance of GRADS is Positively Correlated to Model Layer

To validate the conclusion that the disparity in gradient flow between effective and ineffective demonstrations increases with network layer in Theorem 1, thereby enhancing the demonstration selection capability, we conduct the experiment. We propose to employ GRADS to select demonstrations using gradient flows from different layers for demonstration selection. Considering the experimental cost, we evaluate under each 5 layer.

Dataset	Layer							
	0	5	10	15	20	25	30	31
GSM8K	83.2	84.4	84.9	85.1	85.2	85.6	85.2	**85.6**
MATH	44.4	44.8	45.8	45.2	46.2	47.6	46.8	**48.2**
ARC-Challenge	82.1	82.3	82.1	82.8	83.4	84.0	85.1	**86.3**
MMLU-Pro	50.8	51.5	53.6	54.8	54.8	55.6	**56.2**	56.0

Table 3: The performance of GRADS using the gradient flow of different layer on Llama-3.1-8B. The best result under each setting is marked in **bold**.

The experimental results are presented in Table 3. From the table, we can observe the following: *(i)* Across all datasets, there is a general upward trend in model performance as the number of layers increases, which confirms the conclusion of Theorem 1 that as the network layers increase, the difference in effectiveness among demonstrations is amplified, which enables a better selection of effective demonstrations. *(ii)* However, the performance of GRADS does not monotonically increase with the number of layers, since the models used in our experiments are more complex than those in our theoretical analysis. For instance, the presence of residual streams (He et al., 2016) can diminish the amplifying effect of multi-layer transformers on the gradient flow, which could lead the model to erroneously select ineffective demonstrations, resulting in a decline in performance.

5 Related Works

In-Context Learning ICL can effectively improve LLM reasoning performance by adding query-relevant demonstrations to the prompt with additional training (Brown et al., 2020; Dong et al., 2024). Prior work falls into three axes: synthesizing, selecting, and utilizing demonstrations. To cut labeling cost, LLM-based synthesis draws on existing examples, related information, and similar-task data (Long et al., 2024a; Su et al., 2024; He et al., 2024; Chen et al., 2023b; Wang et al., 2025b; Chen et al., 2025b; Wang et al., 2025a). Selection has progressed from gram-based heuristics to model-aware relevance estimation (Rubin et al., 2022; Li et al., 2023; Peng et al., 2024; Wang et al., 2025e; Yang et al., 2023; Ye et al., 2023; Wang et al., 2024a; Zhang et al., 2025c). Utilization improves effectiveness and reduces inference cost via demonstration reordering or encoding-and-injection (Lu et al., 2022; Pham et al., 2025; Li et al., 2024a; Wang et al., 2025d; Li et al., 2025). Across these strands, a unifying theme is maximizing task alignment under tight context budgets—balancing coverage, diversity, and cost during inference.

However, existing demonstration selection methods mainly consider query–demonstration relevance, overlooking that demonstrations may be ineffective if the model has already learned their information. Thus, we propose GRADS, a gradient-flow–based approach that selects demonstrations contributing significant new information during inference, thereby improving ICL performance.

Mechanism of In-Context Learning Understanding ICL's mechanism guides performance improvements and clarifies model reasoning (Zhou et al., 2024). Prior work falls into four strands: theory, architecture, data, and inference. Theoretical studies establish ICL's effectiveness, like convergence and rates Wies et al. (2023); Huang et al. (2024); Yang et al. (2024); Smart et al. (2025); Fu et al. (2024); Huang et al. (2025); Vladymyrov et al. (2024)—often via LSA-based analyses (Zhang et al., 2024; Mahankali et al., 2024; Lu et al., 2024). Architectural analyses show attention chiefly drives ICL (Olsson et al., 2022; Chen et al., 2024; Oko et al., 2024), while MLPs play a supporting role (Li et al., 2024b; Nguyen and Reddy, 2025). Data-centric studies attribute ICL emergence to task diversity and a training-phase shift from in-weight to in-context learning (Raventos et al., 2023; Wibisono and Wang, 2024; Zhang et al., 2025a; Goddard et al., 2025; Zhang et al., 2025b; Singh et al., 2025; Park et al., 2025; de Wynter, 2025). Inference-time analyses examine observed behaviors, including factor influences and changes in internal computations (Bigelow et al., 2024; Shi et al., 2024; Lin and Lee, 2024; Long et al., 2024b; Sia et al., 2024; Zhao et al., 2024).

However, most studies of the ICL mechanism assume that provided demonstrations are effective. In practice, however, demonstrations can be ineffective, yielding no performance gain (DeepSeek-AI, 2025; Wang et al., 2025c). In this work, we examine this phenomenon by identifying two key factors, including irrelevant content or information already learned by the model, and reveal that multi-layer transformers amplify demonstration effectiveness.

6 Conclusion

In this paper, we first discuss that a demonstration is ineffective because the its information has been learned by the model or is irrelevant to the query. We then demonstrate that as the model layer increases, the ratio of the accumulated gradient flow among demonstrations is amplified. Analytical experiments show that as the model layer increases, the difference in effectiveness among demonstrations is magnified, which corroborates our theoretical derivations. Second, we present GRADS, which achieves an average relative performance improvement of 1.3% compared to existing demonstration selection methods, achieving new SOTA results, which proves the effectiveness of our method.

Limitations

(i) We have not validated GRADS on additional datasets such as Question Answering and Code Generation, and we will include results on these datasets in future work. *(ii)* GRADS cannot be applied to closed-source models, where the gradient flow is not available.

Ethics Statement

All datasets and models used in this paper are publicly available, and our usage follows their licenses and terms. We have employed the LLM tools for coding and polishing.

References

et al Aaron Grattafiori, Abhimanyu Dubey. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

et al An Yang, Anfeng Li. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Eric J Bigelow, Ekdeep Singh Lubana, Robert P. Dick, Hidenori Tanaka, and Tomer Ullman. 2024. In-context learning dynamics with random binary sequences. In *The Twelfth International Conference on Learning Representations*.

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, and 12 others. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122.

Jiuhai Chen, Lichang Chen, Chen Zhu, and Tianyi Zhou. 2023a. How many demonstrations do you need for in-context learning? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11149–11159, Singapore. Association for Computational Linguistics.

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. 2025a. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *Preprint*, arXiv:2503.09567.

Wei-Lin Chen, Cheng-Kuang Wu, Yun-Nung Chen, and Hsin-Hsi Chen. 2023b. Self-ICL: Zero-shot in-context learning with self-generated demonstrations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15651–15662, Singapore. Association for Computational Linguistics.

Xingwu Chen, Lei Zhao, and Difan Zou. 2024. How transformers utilize multi-head attention in in-context learning? a case study on sparse linear regression. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zihan Chen, Song Wang, Zhen Tan, Jundong Li, and Cong Shen. 2025b. MAPLE: Many-shot adaptive pseudo-labeling for in-context learning. In *Forty-second International Conference on Machine Learning*.

Hakaze Cho, Mariko Kato, Yoshihiro Sakai, and Naoya Inoue. 2025. Revisiting in-context learning inference circuit in large language models. In *The Thirteenth International Conference on Learning Representations*.

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. *Preprint*, arXiv:2110.14168.

Adrian de Wynter. 2025. Is in-context learning learning? *Preprint*, arXiv:2509.10414.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128, Miami, Florida, USA. Association for Computational Linguistics.

Deqing Fu, Tian qi Chen, Robin Jia, and Vatsal Sharan. 2024. Transformers learn to achieve second-order convergence rates for in-context linear regression. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Chase Goddard, Lindsay M. Smith, Vudtiwat Ngampruetikorn, and David J. Schwab. 2025. When can in-context learning generalize out of task distribution? In *Forty-second International Conference on Machine Learning*.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Wei He, Shichun Liu, Jun Zhao, Yiwen Ding, Yi Lu, Zhiheng Xi, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Self-demos: Eliciting out-of-demonstration generalizability in large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3829–3845, Mexico City, Mexico. Association for Computational Linguistics.

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 *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Jianhao Huang, Zixuan Wang, and Jason D. Lee. 2025. Transformers learn to implement multi-step gradient descent with chain of thought. In *The Thirteenth International Conference on Learning Representations*.

Yu Huang, Yuan Cheng, and Yingbin Liang. 2024. In-context convergence of transformers.

Janak Kapuriya, Manit Kaushik, Debasis Ganguly, and Sumit Bhatia. 2025. Exploring the role of diversity in example selection for in-context learning. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '25, page 2962–2966, New York, NY, USA. Association for Computing Machinery.

Wonbin Kweon, SeongKu Kang, Runchu Tian, Pengcheng Jiang, Jiawei Han, and Hwanjo Yu. 2025. Topic coverage-based demonstration retrieval for in-context learning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 19911–19923, Suzhou, China. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Dongfang Li, zhenyu liu, Xinshuo Hu, Zetian Sun, Baotian Hu, and Min Zhang. 2024a. In-context learning state vector with inner and momentum optimization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Hongkang Li, Meng Wang, Songtao Lu, Xiaodong Cui, and Pin-Yu Chen. 2024b. How do nonlinear transformers learn and generalize in in-context learning? In *Forty-first International Conference on Machine Learning*.

Xiaonan Li, Kai Lv, Hang Yan, Tianyang Lin, Wei Zhu, Yuan Ni, Guotong Xie, Xiaoling Wang, and Xipeng Qiu. 2023. Unified demonstration retriever for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4644–4668, Toronto, Canada. Association for Computational Linguistics.

Zhuowei Li, Zihao Xu, Ligong Han, Yunhe Gao, Song Wen, Di Liu, Hao Wang, and Dimitris N. Metaxas. 2025. Implicit in-context learning. In *The Thirteenth International Conference on Learning Representations*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Ziqian Lin and Kangwook Lee. 2024. Dual operating modes of in-context learning. In *Forty-first International Conference on Machine Learning*.

Hui Liu, Wenya Wang, Hao Sun, Chris Xing Tian, Chenqi Kong, Xin Dong, and Haoliang Li. 2025. Unraveling the mechanics of learning-based demonstration selection for in-context learning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2623–2641, Vienna, Austria. Association for Computational Linguistics.

Yiting Liu and Zhi-Hong Deng. 2025. Iterative vectors: In-context gradient steering without backpropagation. In *Forty-second International Conference on Machine Learning*.

Lin Long, Rui Wang, Ruixuan Xiao, Junbo Zhao, Xiao Ding, Gang Chen, and Haobo Wang. 2024a. On LLMs-driven synthetic data generation, curation, and evaluation: A survey. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11065–11082, Bangkok, Thailand. Association for Computational Linguistics.

Quanyu Long, Yin Wu, Wenya Wang, and Sinno Jialin Pan. 2024b. Does in-context learning really learn? rethinking how large language models respond and solve tasks via in-context learning. In *First Conference on Language Modeling*.

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)*, pages 8086–8098, Dublin, Ireland. Association for Computational Linguistics.

Yue Lu, Mary Letey, Jacob A Zavatone-Veth, Anindita Maiti, and Cengiz Pehlevan. 2024. In-context learning by linear attention: Exact asymptotics and experiments. In *NeurIPS 2024 Workshop on Mathematics of Modern Machine Learning*.

Arvind V. Mahankali, Tatsunori Hashimoto, and Tengyu Ma. 2024. One step of gradient descent is provably the optimal in-context learner with one layer of linear self-attention. In *The Twelfth International Conference on Learning Representations*.

Alex Nguyen and Gautam Reddy. 2025. Differential learning kinetics govern the transition from memorization to generalization during in-context learning. In *The Thirteenth International Conference on Learning Representations*.

Tai Nguyen and Eric Wong. 2023. In-context example selection with influences. *Preprint*, arXiv:2302.11042.

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 188–197, Hong Kong, China. Association for Computational Linguistics.

Kazusato Oko, Yujin Song, Taiji Suzuki, and Denny Wu. 2024. Pretrained transformer efficiently learns low-dimensional target functions in-context. In *NeurIPS*.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, and 7 others. 2022. In-context learning and induction heads. *Preprint*, arXiv:2209.11895.

Core Francisco Park, Ekdeep Singh Lubana, and Hidenori Tanaka. 2025. Competition dynamics shape algorithmic phases of in-context learning. In *The Thirteenth International Conference on Learning Representations*.

Keqin Peng, Liang Ding, Yancheng Yuan, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2024. Revisiting demonstration selection strategies in in-context learning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9090–9101, Bangkok, Thailand. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Kha Pham, Hung Le, Man Ngo, and Truyen Tran. 2025. Rapid selection and ordering of in-context demonstrations via prompt embedding clustering. In *The Thirteenth International Conference on Learning Representations*.

Allan Raventos, Mansheej Paul, Feng Chen, and Surya Ganguli. 2023. Pretraining task diversity and the emergence of non-bayesian in-context learning for regression. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2655–2671, Seattle, United States. Association for Computational Linguistics.

Zhenmei Shi, Junyi Wei, Zhuoyan Xu, and Yingyu Liang. 2024. Why larger language models do in-context learning differently? In *Forty-first International Conference on Machine Learning*.

Suzanna Sia, David Mueller, and Kevin Duh. 2024. Where does in-context learning \\ happen in large language models? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Aaditya K Singh, Ted Moskovitz, Sara Dragutinović, Felix Hill, Stephanie C.Y. Chan, and Andrew M Saxe. 2025. Strategy coopetition explains the emergence and transience of in-context learning. In *Forty-second International Conference on Machine Learning*.

Matthew Smart, Alberto Bietti, and Anirvan M. Sengupta. 2025. In-context denoising with one-layer transformers: Connections between attention and associative memory retrieval. In *Forty-second International Conference on Machine Learning*.

Yi Su, Yunpeng Tai, Yixin Ji, Juntao Li, Yan Bowen, and Min Zhang. 2024. Demonstration augmentation for zero-shot in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14232–14244, Bangkok, Thailand. Association for Computational Linguistics.

Wenbiao Tao, Hanlun Zhu, Keren Tan, Jiani Wang, Yuanyuan Liang, Huihui Jiang, Pengcheng Yuan, and Yunshi Lan. 2024. Finqa: A training-free dynamic knowledge graph question answering system in finance with llm-based revision. In *Machine Learning and Knowledge Discovery in Databases. Research Track and Demo Track: European Conference, ECML PKDD 2024, Vilnius, Lithuania, September 9–13, 2024, Proceedings, Part VIII*, page 418–423, Berlin, Heidelberg. Springer-Verlag.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Max Vladymyrov, Johannes von Oswald, Mark Sandler, and Rong Ge. 2024. Linear transformers are versatile in-context learners. In *ICML 2024 Workshop on In-Context Learning*.

Dingzirui Wang, Xuanliang Zhang, Qiguang Chen, Longxu Dou, Xiao Xu, Rongyu Cao, YINGWEI MA, Qingfu Zhu, Wanxiang Che, Binhua Li, Fei Huang, and Yongbin Li. 2025a. In-context transfer learning: Demonstration synthesis by transferring similar tasks.

Dingzirui Wang, Xuanliang Zhang, Keyan Xu, Qingfu Zhu, Wanxiang Che, and Yang Deng. 2025b. V-synthesis: Task-agnostic synthesis of consistent and diverse in-context demonstrations from scratch via v-entropy. *Preprint*, arXiv:2506.23149.

Dingzriui Wang, Xuanliang Zhang, Keyan Xu, Qingfu Zhu, Wanxiang Che, and Yang Deng. 2025c. Learning-to-context slope: Evaluating in-context learning effectiveness beyond performance illusions. *Preprint*, arXiv:2506.23146.

Futing Wang, Jianhao Yan, Yue Zhang, and Tao Lin. 2025d. ELICIT: LLM augmentation via external in-context capability. In *The Thirteenth International Conference on Learning Representations*.

Lean Wang, Lei Li, Damai Dai, Deli Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. 2023. Label words are anchors: An information flow perspective for understanding in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9840–9855, Singapore. Association for Computational Linguistics.

Song Wang, Zihan Chen, Chengshuai Shi, Cong Shen, and Jundong Li. 2024a. Mixture of demonstrations for in-context learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Xubin Wang, Jianfei Wu, Yuan Yichen, Deyu Cai, Mingzhe Li, and Weijia Jia. 2025e. Demonstration selection for in-context learning via reinforcement learning. In *Forty-second International Conference on Machine Learning*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2024b. MMLU-pro: A more robust and challenging multi-task language understanding benchmark. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and Tengyu Ma. 2024. Larger language models do in-context learning differently.

Kevin Christian Wibisono and Yixin Wang. 2024. From unstructured data to in-context learning: Exploring what tasks can be learned and when. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Noam Wies, Yoav Levine, and Amnon Shashua. 2023. The learnability of in-context learning. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yuan Wu, Diana Inkpen, and Ahmed El-Roby. 2022. Maximum batch frobenius norm for multi-domain text classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23-27 May 2022*, pages 3763–3767. IEEE.

Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2019. Quick and (not so) dirty: Unsupervised selection of justification sentences for multi-hop question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2578–2589, Hong Kong, China. Association for Computational Linguistics.

Tong Yang, Yu Huang, Yingbin Liang, and Yuejie Chi. 2024. In-context learning with representations: Contextual generalization of trained transformers. In *ICML 2024 Workshop on Theoretical Foundations of Foundation Models*.

Zhao Yang, Yuanzhe Zhang, Dianbo Sui, Cao Liu, Jun Zhao, and Kang Liu. 2023. Representative demonstration selection for in-context learning with two-stage determinantal point process. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5443–5456, Singapore. Association for Computational Linguistics.

Xi Ye, Srinivasan Iyer, Asli Celikyilmaz, Veselin Stoyanov, Greg Durrett, and Ramakanth Pasunuru. 2023. Complementary explanations for effective in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4469–4484,

Toronto, Canada. Association for Computational Linguistics.

Ruiqi Zhang, Spencer Frei, and Peter L. Bartlett. 2024. Trained transformers learn linear models in-context. *Journal of Machine Learning Research*, 25(49):1–55.

Xingxuan Zhang, Haoran Wang, Jiansheng Li, Yuan Xue, Shikai Guan, Renzhe Xu, Hao Zou, Han Yu, and Peng Cui. 2025a. Understanding the generalization of in-context learning in transformers: An empirical study. In *The Thirteenth International Conference on Learning Representations*.

Yedi Zhang, Aaditya K Singh, Peter E. Latham, and Andrew M Saxe. 2025b. Training dynamics of in-context learning in linear attention. In *Forty-second International Conference on Machine Learning*.

Zheng Zhang, Shaocheng Lan, Lei Song, Jiang Bian, Yexin Li, and Kan Ren. 2025c. Learning to select in-context demonstration preferred by large language model. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11345–11360, Vienna, Austria. Association for Computational Linguistics.

Ziniu Zhang, Zhenshuo Zhang, Dongyue Li, Lu Wang, Jennifer Dy, and Hongyang R. Zhang. 2025d. Linear-time demonstration selection for in-context learning via gradient estimation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 16470–16488, Suzhou, China. Association for Computational Linguistics.

Siyan Zhao, Tung Nguyen, and Aditya Grover. 2024. Probing the decision boundaries of in-context learning in large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yuxiang Zhou, Jiazheng Li, Yanzheng Xiang, Hanqi Yan, Lin Gui, and Yulan He. 2024. The mystery of in-context learning: A comprehensive survey on interpretation and analysis. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14365–14378, Miami, Florida, USA. Association for Computational Linguistics.

A Proof

A.1 Proof of Equation 2

Proof. Based on Equation 1 and Zhang et al. (2024), the predicted answer of LSA is:

$$\hat{y}_{query} = \begin{pmatrix} (w_{21}^{PV})^\top & w_{22}^{PV} \end{pmatrix} \cdot \left(\frac{EE^\top}{N} \right) \cdot \begin{pmatrix} W_{11}^{KQ} \\ (w_{21}^{KQ})^\top \end{pmatrix} q_x, \tag{7}$$

where N is the number of demonstrations. For $N = 1$, the matrix product $EE^\top$ can be expressed compactly as $EE^\top = dd^\top + qq^\top$. Substituting this into the equation gives:

$$\hat{y}_{query} = \begin{pmatrix} (w_{21}^{PV})^\top & w_{22}^{PV} \end{pmatrix} (dd^\top + qq^\top) \begin{pmatrix} W_{11}^{KQ} \\ (w_{21}^{KQ})^\top \end{pmatrix} q_x \tag{8}$$

We can expand this expression and separate the terms that depend on d:

$$\hat{y}_{query} = \underbrace{\begin{pmatrix} (w_{21}^{PV})^\top & w_{22}^{PV} \end{pmatrix} (dd^\top) \begin{pmatrix} W_{11}^{KQ} \\ (w_{21}^{KQ})^\top \end{pmatrix} q_x}_{\text{Term depending on } d} + \underbrace{\begin{pmatrix} (w_{21}^{PV})^\top & w_{22}^{PV} \end{pmatrix} (qq^\top) \begin{pmatrix} W_{11}^{KQ} \\ (w_{21}^{KQ})^\top \end{pmatrix} q_x}_{\text{Constant w.r.t. } d}$$

The term depending on d can be rewritten using the property of scalar products:

$$\begin{pmatrix} (w_{21}^{PV})^\top & w_{22}^{PV} \end{pmatrix} dd^\top \begin{pmatrix} W_{11}^{KQ} \\ (w_{21}^{KQ})^\top \end{pmatrix} q_x = \left(\begin{pmatrix} (w_{21}^{PV})^\top & w_{22}^{PV} \end{pmatrix} d \right) \left(d^\top \begin{pmatrix} W_{11}^{KQ} \\ (w_{21}^{KQ})^\top \end{pmatrix} q_x \right)$$

This is a quadratic form of the vector d. To compute its gradient, we use the vector calculus identity $\nabla_z((a^\top z)(b^\top z)) = a(b^\top z) + b(a^\top z)$. We set:

$$a = \begin{pmatrix} w_{21}^{PV} \\ w_{22}^{PV} \end{pmatrix}$$

$$b = \begin{pmatrix} W_{11}^{KQ} q_x \\ (w_{21}^{KQ})^\top q_x \end{pmatrix}$$

The second term in the expression for $\hat{y}_{query}$ is constant with respect to d, so its gradient is zero. Applying the identity to the first term yields the gradient of $\hat{y}_{query}$ with respect to d:

$$\begin{aligned} \nabla_d \hat{y}_{query} &= \begin{pmatrix} w_{21}^{PV} \\ w_{22}^{PV} \end{pmatrix} \left(\begin{pmatrix} d_x \\ d_y \end{pmatrix}^\top \begin{pmatrix} W_{11}^{KQ} q_x \\ (w_{21}^{KQ})^\top q_x \end{pmatrix} \right) \\ &+ \begin{pmatrix} W_{11}^{KQ} q_x \\ (w_{21}^{KQ})^\top q_x \end{pmatrix} \left(\begin{pmatrix} w_{21}^{PV} \\ w_{22}^{PV} \end{pmatrix}^\top \begin{pmatrix} d_x \\ d_y \end{pmatrix} \right) \\ &= \begin{pmatrix} w_{21}^{PV} \\ w_{22}^{PV} \end{pmatrix} \left(d_x^\top W_{11}^{KQ} q_x + d_y (w_{21}^{KQ})^\top q_x \right) \\ &+ \begin{pmatrix} W_{11}^{KQ} q_x \\ (w_{21}^{KQ})^\top q_x \end{pmatrix} \left((w_{21}^{PV})^\top d_x + d_y w_{22}^{PV} \right) \end{aligned}$$

This completes the derivation. □

A.2 Proof of Lemma 1

Proof. We prove by induction on the layer index l.

Base case ($l = 0$). The two inequalities in the statement are satisfied by assumption.

Induction step. Assume the inequalities hold for some layer $l - 1$ ($1 \le l \le L$), i.e.

$$\begin{aligned} &\left\| W^{PV,(l-1)} d_1^{(l-1)} \right\| \ge \left\| W^{PV,(l-1)} d_2^{(l-1)} \right\| \\ &\left\| \left(d_1^{(l-1)} \right)^\top W^{KQ,(l-1)} q^{(l-1)} \right\| \ge \\ &\left\| \left(d_2^{(l-1)} \right)^\top W^{KQ,(l-1)} q^{(l-1)} \right\|. \end{aligned}$$

Apply (5) to both demonstrations and use the strict monotonicity of g_l:

$$\begin{aligned} &\left\| W^{PV,(l)} d_1^{(l)} \right\| \\ &= g_l\left(\left\| W^{PV,(l-1)} d_1^{(l-1)} \right\| \right) \\ &\ge g_l\left(\left\| W^{PV,(l-1)} d_2^{(l-1)} \right\| \right) \\ &= \left\| W^{PV,(l)} d_2^{(l)} \right\|. \end{aligned}$$

Similarly, apply (6) and the strict monotonicity of h_l:

$$\begin{aligned} &\left\| \left(d_1^{(l)} \right)^\top W^{KQ,(l)} q^{(l)} \right\| \\ &= h_l\left(\left\| \left(d_1^{(l-1)} \right)^\top W^{KQ,(l-1)} q^{(l-1)} \right\| \right) \\ &\ge h_l\left(\left\| \left(d_2^{(l-1)} \right)^\top W^{KQ,(l-1)} q^{(l-1)} \right\| \right) \\ &= \left\| \left(d_2^{(l)} \right)^\top W^{KQ,(l)} q^{(l)} \right\|. \end{aligned}$$

Thus the claim holds for layer l. By induction, it holds for all layers $l = 0, \ldots, L$. □

Prompt of Inference
{task} Below are some examples — {demo} — Based on the above instruction and examples, solve the following problem. {question}

Table 4: The prompt of the inference.

A.3 Proof of Theorem 1

Proof. Based on Equation 4, we can derive:

$$\frac{\partial \hat{q}_y^{(l_1)}}{\partial d^{(0)}} \Big/ \frac{\partial \hat{q}_y^{(l_2)}}{\partial d^{(0)}}$$
$$= \left(\frac{\partial \hat{q}_y^{(l_1)}}{\partial E^{(l_1-1)}} \times \frac{\partial E^{(1)}}{\partial d^{(0)}} \times \prod_{i=2}^{l_1} \frac{\partial E^{(i)}}{\partial E^{(i-1)}} \right) \Big/ \left(\frac{\partial \hat{q}_y^{(l_2)}}{\partial E^{(l_2-1)}} \times \frac{\partial E^{(1)}}{\partial d^{(0)}} \times \prod_{i=2}^{l_2} \frac{\partial E^{(i)}}{\partial E^{(i-1)}} \right)$$
$$= \left(\frac{\partial \hat{q}_y^{(l_1)}}{\partial E^{(l_1-1)}} \Big/ \frac{\partial \hat{q}_y^{(l_2)}}{\partial E^{(l_2-1)}} \right) \times \prod_{i=l_2+1}^{l_1} \frac{\partial E^{(i)}}{\partial E^{(i-1)}}$$

Consider the total derivative of f_{LSA}:

$$\begin{aligned} \mathrm{d}f_{\mathrm{LSA}} =& \mathrm{d}E + W^{PV}\,\mathrm{d}E\,E^\top W^{KQ} E + \\ & W^{PV} E \big(E^\top W^{KQ}\,\mathrm{d}E + \mathrm{d}E^\top W^{KQ} E \big) \end{aligned}$$

It can be observed that the coefficients of the differential terms are consistent with those in Definition 1. Therefore, from $d_1^{(l)} \succcurlyeq_{q^{(l)};\theta^{(l)}} d_2^{(l)}$, we know that:

$$\frac{\partial E^{(l)}}{\partial E^{(l-1)}}(E_1;\theta) \geq \frac{\partial E^{(l)}}{\partial E^{(l-1)}}(E_2;\theta),$$

where $l \in \{l_2+1, \ldots, l_1\}$. Therefore, we can conclude that:

$$\frac{\|\partial_{d^{(0)}} \hat{q}_y{}^{(l_1)}(E_1;\theta)\|}{\|\partial_{d^{(0)}} \hat{q}_y{}^{(l_1)}(E_2;\theta)\|} \geq \frac{\|\partial_{d^{(0)}} \hat{q}_y{}^{(l_2)}(E_1;\theta)\|}{\|\partial_{d^{(0)}} \hat{q}_y{}^{(l_2)}(E_2;\theta)\|}$$

□

B Additional Information

B.1 Prompt

In this section, we present the inference prompt of our main experiment, as shown in Table 4. The task definition we used is the same as the previous works (DeepSeek-AI, 2025; Aaron Grattafiori, 2024).

Dataset	Test Set	Demonstration
GSM8K	1319	7473
MATH	500	7496
ARC-Challenge	1172	1119
MMLU-Pro	1000	70
Amazon Review	200	1800
FinQA	6251	1147

Table 5: The scales of test set and demonstrations of each dataset.

B.2 Dataset

In this section, we detail the datasets used in our study. Table 5 summarizes the scale of the test set and demonstrations for each.

GSM8K GSM8K (Cobbe et al., 2021) is a high-quality collection of elementary school-level math problems. We utilize its training set directly as the demonstration pool.

MATH The MATH dataset (Hendrycks et al., 2021) consists of challenging high school competition-level math problems in fields like algebra, probability, and geometry. Following the approach of Lightman et al. (2024), we evaluate GRADS on a random sample of 500 problems. The demonstrations are drawn from the official training set.

ARC-Challenge The ARC-Challenge (Yadav et al., 2019) is a question-answering dataset with difficult, science-focused questions. For this dataset, the training set is used as our demonstration pool.

MMLU-Pro MMLU-Pro (Wang et al., 2024b) serves as a multi-task benchmark for the comprehensive evaluation of LLMs on professional domain knowledge and complex reasoning. As the dataset is only divided into validation and test sets, we use the validation set as our demonstration pool and conduct evaluations on the test set.

Amazon Review The Amazon Review dataset (Ni et al., 2019), containing a vast amount of user ratings and reviews, is widely used for research in sentiment analysis and recommender systems. Due to the immense size, we select the *Health and Personal Care* category for testing, while using the *All Beauty*, *Digital Music*, and *Software* categories to form the demonstrations.

FinQA FinQA (Tao et al., 2024) is a large-scale benchmark for numerical reasoning in fi-

nance. It contains 8,281 expert-written question–answer pairs built from S&P 500 companies' earnings/annual reports, where evidence can come from both narrative text and structured tables. For each question, FinQA provides supporting facts and an annotated reasoning program that can be executed to obtain the answer, enabling fully explainable evaluation.

B.3 Baseline

BM25 BM25 is a classic sparse retrieval method based on the probabilistic relevance framework, serving as an extension of TF-IDF. In the context of ICL, it treats the test query as a search query and the candidate demonstrations as documents. It ranks demonstrations by scoring them based on the frequency and distribution of query terms, without considering word order or deep semantics. The top-K scored demonstrations are then selected as the in-context demonstrations. This "bag-of-words" approach is known for its computational efficiency and serves as a strong baseline.

MMR Maximal Marginal Relevance (MMR) is a selection strategy designed to balance the relevance of demonstrations to the query with the diversity within the selected set. The rationale is that selecting only the nearest neighbors (most relevant examples) can result in a set of overly similar demonstrations, which may limit the variety of reasoning processes shown to the model. MMR addresses this by iteratively selecting demonstrations that maximize a combined score of relevance to the query and dissimilarity from the examples already chosen. This approach aims to create a set of demonstrations that are not only relevant but also complementary, using diversity as a proxy for complementarity to improve ICL performance.

MoD Mixture of Demonstrations (MoD) is a framework designed to overcome the challenges of a large search space and suboptimal retriever optimization in ICL demonstration selection. The core idea is to partition the entire demonstration pool into distinct groups, typically using K-means clustering on sentence embeddings. Each group is governed by a dedicated "expert", a unique retriever model trained specifically for that partition. During inference, these experts collaboratively retrieve demonstrations for a given query, with the final set being an aggregation of examples selected by the most relevant experts. This "mixture of experts" approach reduces the search complexity while ensuring the selected demonstrations are diverse and effective.

Influence Influence (Nguyen and Wong, 2023) treats demonstration quality as the causal contribution an example makes to in context learning performance. It repeatedly samples many random subsets of size k from the training set, places each subset in the prompt, and evaluates the model on a validation set to obtain a score. For a candidate example i, it estimates in context influence as the average score of subsets that include i minus the average score of subsets that exclude i. It then ranks examples by this influence and selects the top k positive ones, while also revealing harmful examples and order effects such as recency bias.

DICL DICL (Kapuriya et al., 2025) aims to balance relevance and diversity when choosing demonstrations. It first retrieves a candidate pool larger than k using sparse or dense similarity, such as TF IDF or sentence embeddings, so there is room to diversify within a locally relevant neighborhood. It then applies a greedy Maximum Marginal Relevance style reranking that iteratively picks examples that are highly similar to the input while remaining dissimilar to already selected demonstrations. A weight parameter lambda controls the trade off between relevance and diversity. This reduces topical redundancy from pure similarity retrieval and tends to yield more stable gains across context sizes and similarity functions.

GenICL GenICL (Zhang et al., 2025c) frames demonstration selection as learning which examples an LLM *prefers* to see in the prompt, rather than optimizing a surrogate similarity objective. For each training query, it uses LLM feedback to assign a preference score to candidate demonstrations (e.g., how well the LLM predicts the gold output when conditioned on that single demonstration), and forms preferred vs. non-preferred demonstration pairs. It then trains a generative preference model with a latent variable that bridges selection and inference, optimizing a preference-learning objective (with regularization to avoid drifting too far from a reference behavior). At inference time, GenICL first narrows the search space with an off-the-shelf retriever, and then ranks candidates by the learned probability of generating the latent "preferred demonstration" variable for the test input. The top-K candidates under this learned preference score are used as the in-context demonstra-

tions, aiming to select examples that are directly aligned with downstream ICL utility rather than surface similarity.

TopicK TopicK (Kweon et al., 2025) selects demonstrations by explicitly maximizing *topic coverage* of the fine-grained knowledge required by the test input, instead of relying only on embedding similarity. It builds a global topic set from the demonstration pool and trains a lightweight topic predictor to estimate (i) the required topic distribution of a test input and (ii) the covered topic distribution of each candidate demonstration. In addition, TopicK estimates the model's topical knowledge (a prior over topics the LLM already "knows well") using aggregated zero-shot performance, and down-weights such topics during retrieval. Demonstrations are then chosen iteratively with a greedy coverage strategy: each new example is selected to introduce previously uncovered required topics, especially those where the model exhibits low topical knowledge. This yields a set that is both relevant and non-redundant at the topic level, improving complementarity beyond nearest-neighbor retrieval.

ICL-Grad ICL-Grad (Zhang et al., 2025d) accelerates demonstration selection by estimating how candidate examples affect ICL performance using gradients in the *input embedding space*. The method pre-computes model outputs and gradients for an anchor prompt, and then applies a first-order (Taylor) approximation to predict the loss of many different demonstration subsets *without* running full LLM inference for each subset. By sampling many random candidate subsets and estimating their losses cheaply, it aggregates these subset outcomes into an influence-style score for each demonstration (e.g., the average estimated loss over subsets that include that example). The final k demonstrations are selected as those with the most beneficial scores (typically the lowest estimated losses), enabling scalable subset selection with near-linear cost after one gradient pass. Compared with similarity-only retrieval, this approach better reflects the model's behavior under ICL while remaining computationally efficient for large candidate pools.

B.4 Scale of Effective and Ineffective Data

The number of effective and ineffective demonstrations we sampled is shown in Table 6.

C Additional Discussion

C.1 Efficiency of GRADS

The computational cost of GRADS is mainly divided into two parts: the offline computation of the encoding result for each demonstration and the online retrieval of the demonstration and subsequent inference for a query. Let $D = \{d_1, \ldots, d_n\}$ represent the entire demonstration pool, and let $\mathcal{M}_\theta(x)$ denote the computational cost of the model $\mathcal{M}_\theta$ for a given input x. For a query q, let the retrieved demonstration be d_q.

The time complexity of the offline processing for GRADS is: $O(\sum_{i=1}^{n} \mathcal{M}_\theta(d_i))$. Although the offline cost is relatively high, the pre-computation of demonstration encodings is done offline and thus does not affect the online user query process.

The time complexity for online processing is:

$$O(\mathcal{M}_\theta(q) + 4e^2 + \mathcal{M}_\theta(q + d_q)) \tag{9}$$

In Equation 9, the first term represents encoding the user query, the second term corresponds to calculating Equation 2, and the third term is for generating the answer to the user query based on the retrieved demonstration. Considering that $\mathcal{M}_\theta$ is positively correlated with the input length and the complexity of calculating Equation 2 is significantly lower than that of a full model inference $\mathcal{M}_\theta$, the online processing time complexity simplifies to:

$$O(\mathcal{M}_\theta(q + d_q)) \tag{10}$$

This is equivalent to the time complexity of direct 1-shot inference, which demonstrates the high efficiency of GRADS. We also compare the run time with the existing demonstration selection methods in Appendix D.3.

C.2 Effect of Non-Linear Factor of Transformer

In this part, we discuss how different non-linear factors of the Transformer affect the conclusions of this paper.

Softmax attention. While monotone in the alignment score $d^\top W^{KQ} q$, softmax introduces competitive normalization that sharpens high-logit tokens and compresses marginal gradients for medium/low scores. Consequently, demonstrations with high alignment but little novel value information are further down-weighted; with depth, saturation can dampen inter-layer amplification unless the distribution is sufficiently diffuse (e.g., higher temperature).

Model	Type	GSM8K	MATH	ARC-Challenge	MMLU-Pro	Amazon	FinQA
Llama-3.1-8B	Effective	84	52	86	174	26	92
	Ineffective	210	280	212	466	63	267
Llama-R1-8B	Effective	41	27	77	203	20	83
	Ineffective	519	146	198	550	102	202
Qwen3-8B	Effective	33	31	28	251	23	37
	Ineffective	99	111	110	368	77	107

Table 6: The number of effective and ineffective demonstrations under each setting.

FFN activations and LayerNorm. Nonlinear gates (e.g., GELU/SwiGLU) truncate or saturate the propagated value magnitude $\|W^{PV}d\|$, while LayerNorm projects representations onto a sphere, partially washing out norm-based cues. Hence, advantages due to "unassimilated information" are reduced unless coupled with strong alignment; amplification persists when layers operate in near-linear regimes but degrades under saturation.

Residual connections. Residual pathways bypass nonlinear transforms and carry forward prior states, diluting the relative gain of newly injected information along the main branch. This weakens strictly monotonic amplification with depth and explains the non-monotonic, layer-dependent optima observed in practice.

C.3 Why Gradient Flow Can Reflect the Effectiveness

As discussed in Wang et al. (2023), for irrelevant demonstrations, LLMs actively ignore their information, which leads to smaller gradient flow. In other words, the model only produces large gradient flow for effective information, which is consistent with the way attention weights are computed. The experimental results in Table 2 are also consistent with this observation, where the effectiveness of one demonstration is positively correlated with the magnitude of the gradient flow.

D Additional Experiment

D.1 Performance under Multiple Runs

To verify that GRADS consistently enhances ICL performance, we conduct experiments with multiple runs. We use Llama-3.1-8B as the experimental model and perform 5 trials for each setting. As shown in Table 7, GRADS outperforms the baselines across all settings, demonstrating stable improvements in ICL performance.

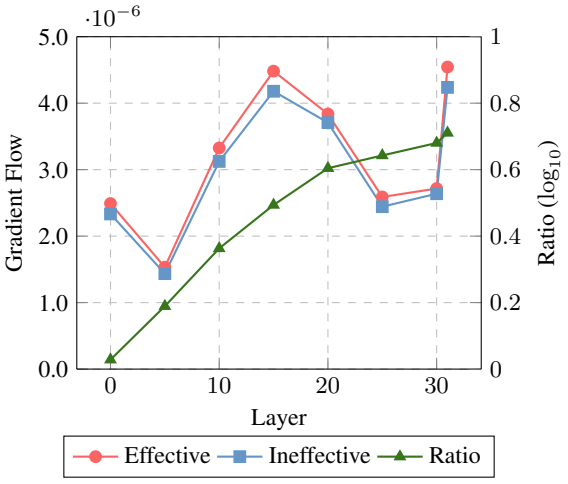

Figure 4: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-3.1-8B on GSM8K.

D.2 Detailed Gradient Flow under Each Setting

In this part, we present the gradient flow under each setting from Figure 4 to Figure 14.

D.3 Run Time Comparison Cross Different Methods

In this section, we compare the run-time efficiency of different demonstration selection methods. Using the same device, we calculate the average time required for demonstration selection per question on the MATH dataset using Llama 3.1-8B. The results are shown in Table 8. We observe that GRADS achieves superior performance with comparable computational efficiency, demonstrating the high efficiency of our method.

D.4 Ablation Study

In this section, we present ablation studies of GRADS to validate the effectiveness of each component in our approach. The results are shown in Table 9, from which we observe that: *(i)* Removing Similarity causes a noticeable drop on MMLU-Pro (-4.0) and ARC-C/FinQA (-2.2 each),

GSM8K	MATH	ARC-C	MMLU-Pro	Amazon	FinQA
85.6 ± 1.4	48.2 ± 2.2	86.3 ± 1.3	56.0 ± 2.0	70.0 ± 1.8	51.1 ± 1.3

Table 7: The performance of GRADS using Llama-3.1-8B on different datasets with five runs.

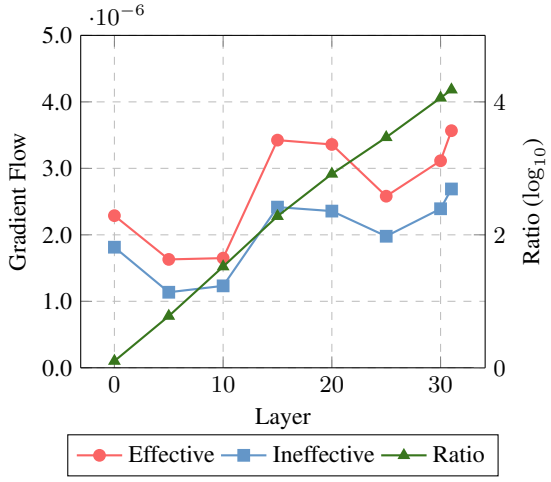

Figure 5: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-3.1-8B on MATH.

Method	EM	T (s)
Zero	47.0	–
BM25	44.6	0.02
MMR	45.2	0.13
MoD	47.0	0.21
Influence	47.4	0.27
DICL	47.4	0.43
GenICL	47.6	0.96
TopicK	47.0	0.83
ICl-Grad	47.0	0.59
GRADS	48.2	0.32

Table 8: The performance and run time of each method on MATH using Llama-3.1-8B. T denotes the average run time of each question.

suggesting that similarity-based signals are particularly important for knowledge-intensive reasoning and multi-step decision tasks, while having only marginal impact on GSM8K (-0.6) and MATH (-0.2). *(ii)* Removing Information leads to the largest degradation on Amazon (-3.5) and ARC-C (-3.4), indicating that the information component contributes more to domain- and context-sensitive understanding, whereas its effect is relatively small on MATH/FinQA (-0.6 each) compared with the similarity component.

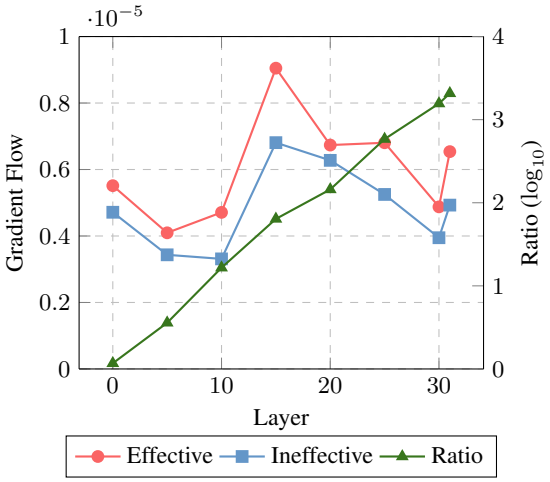

Figure 6: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-3.1-8B on ARC-Challenge.

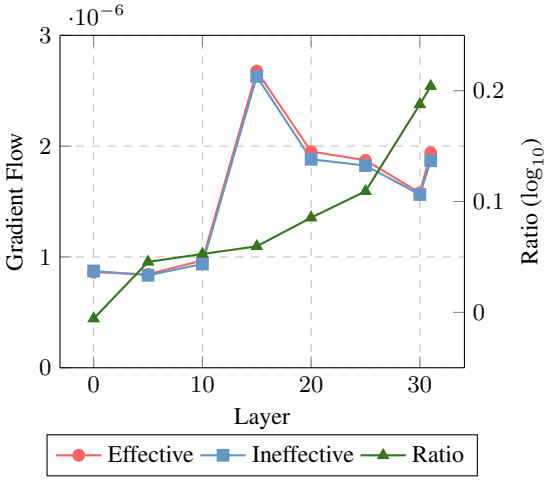

Figure 7: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-3.1-8B on MMLU-Pro.

Method	GSM8K	MATH	ARC-C	MMLU-Pro	Amazon	FinQA
GRADS	85.6	48.2	86.3	56.0	70.0	52.3
- Similarity	$85.0_{(-0.6)}$	$48.0_{(-0.2)}$	$84.1_{(-2.2)}$	$52.0_{(-4.0)}$	$68.0_{(-2.0)}$	$50.1_{(-2.2)}$
- Information	$84.2_{(-1.4)}$	$47.6_{(-0.6)}$	$82.9_{(-3.4)}$	$53.5_{(-2.5)}$	$66.5_{(-3.5)}$	$51.7_{(-0.6)}$

Table 9: The ablation study of GRADS using Llama-3.1-8B. "- Similarity" denotes removing $d^\top W^{KQ} q$. "- Information" denotes removing $W^{(PV)} d$.

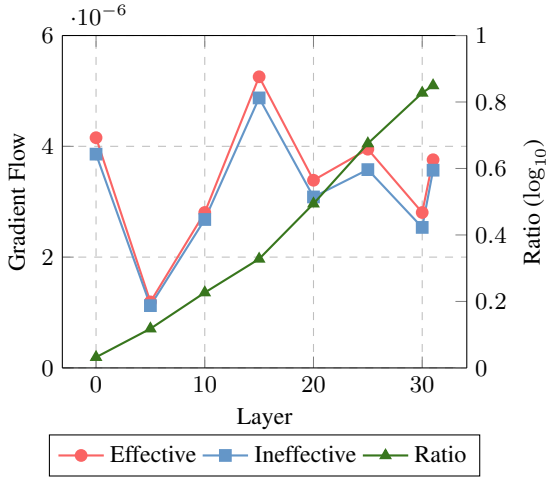

Figure 8: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-3.1-8B on Amazon Review.

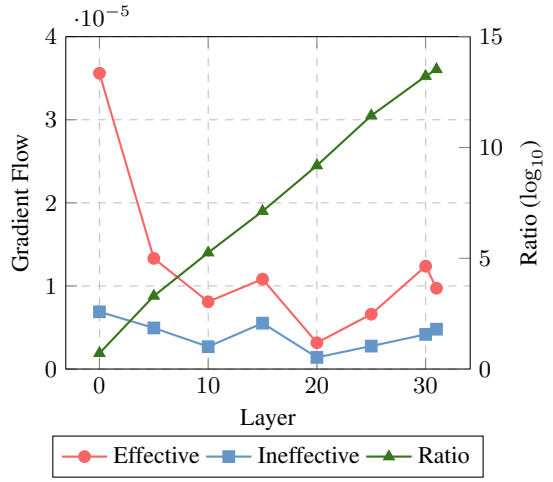

Figure 9: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-R1-8B on GSM8K.

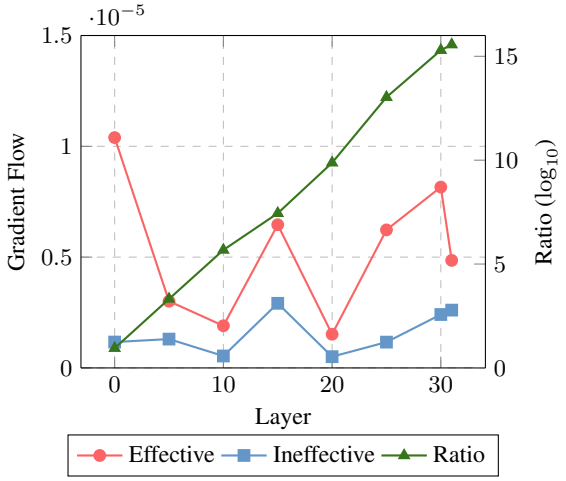

Figure 10: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-R1-8B on MATH.

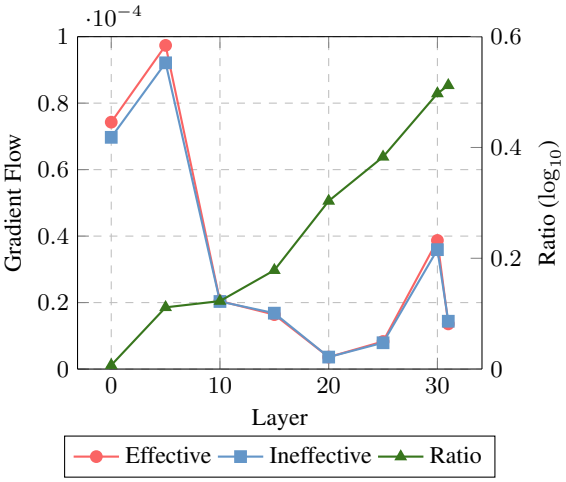

Figure 11: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-R1-8B on ARC-Challenge.

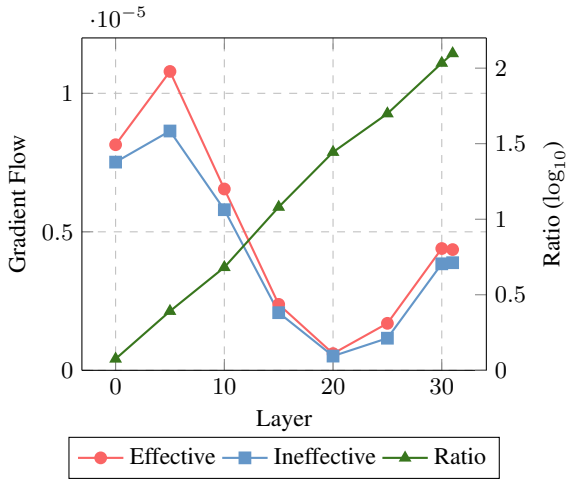

Figure 12: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-R1-8B on MMLU-Pro.

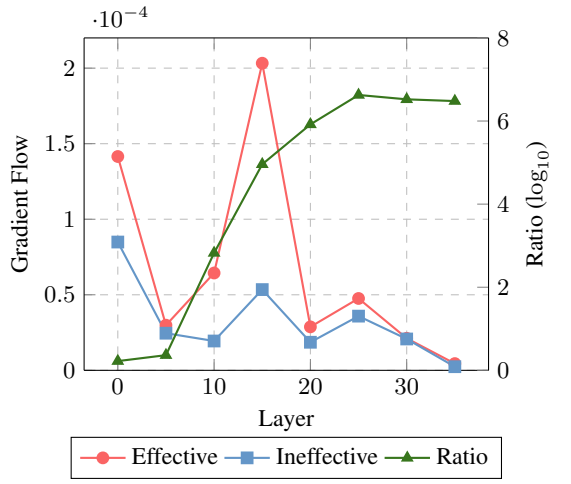

Figure 15: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Qwen3-8B on MATH.

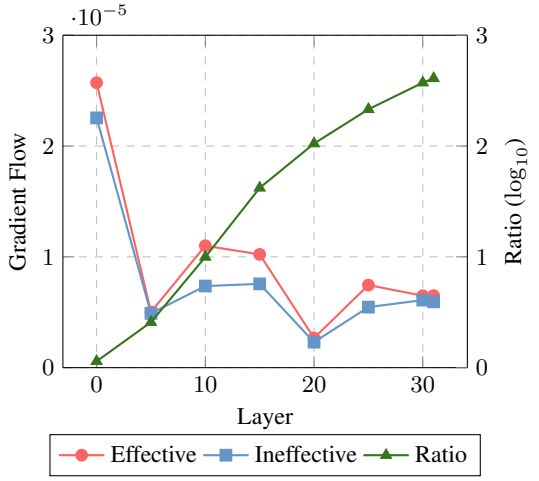

Figure 13: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Llama-R1-8B on Amazon Review.

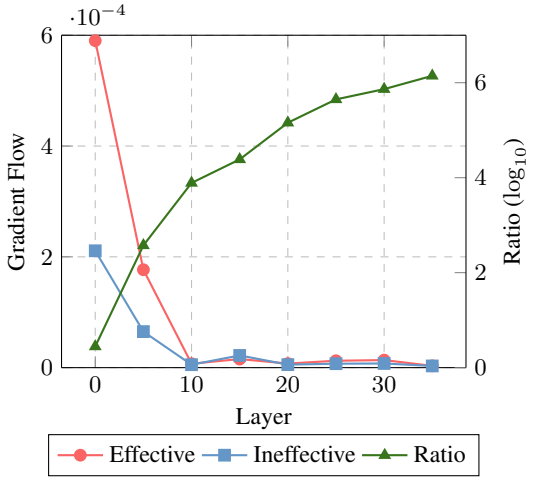

Figure 16: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Qwen3-8B on ARC-Challenge.

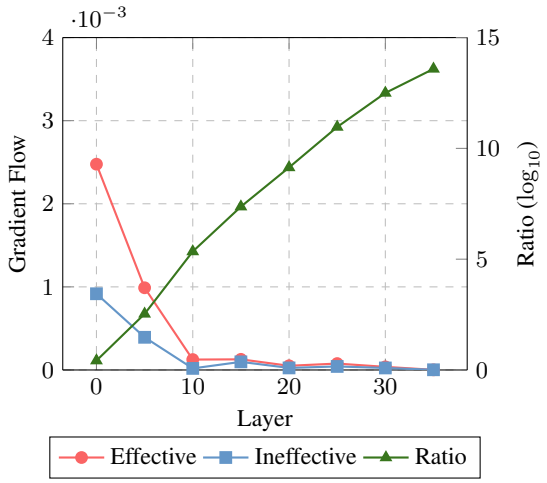

Figure 14: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Qwen3-8B on GSM8K.

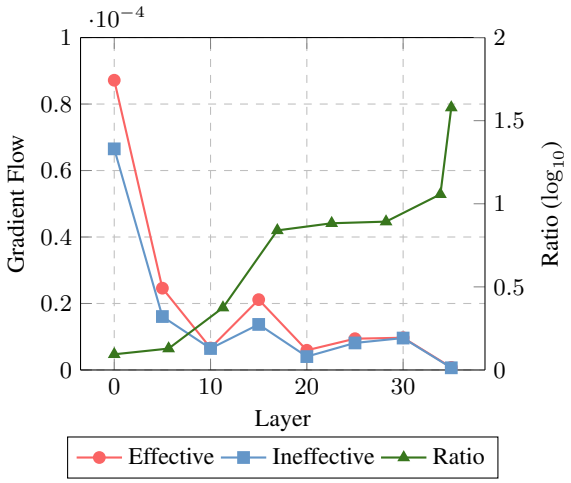

Figure 17: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Qwen3-8B on MMLU-Pro.

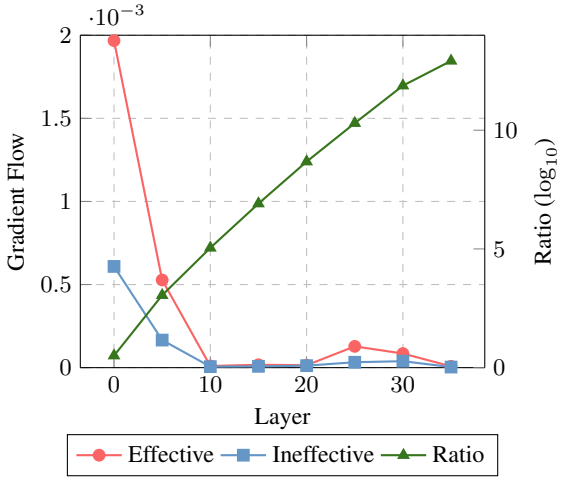

Figure 18: The average gradient flow (left y-axis) and the ratio between the effective and the ineffective (right y-axis) of Qwen3-8B on Amazon Review.