

Revisiting Demonstration Selection Strategies in In-Context Learning

Keqin Peng¹, Liang Ding^{2*}, Yancheng Yuan^{3*}

Xuebo Liu⁴, Min Zhang⁴, Yuanxin Ouyang¹, Dacheng Tao⁵

¹Beihang University ²The University of Sydney ³The Hong Kong Polytechnic University

⁴Harbin Institute of Technology, Shenzhen ⁵Nanyang Technological University

keqin.peng@buaa.edu.cn, liangding@gmail.com

Abstract

Large language models (LLMs) have shown an impressive ability to perform a wide range of tasks using in-context learning (ICL), where a few examples are used to describe a task to the model. However, the performance of ICL varies significantly with the choice of demonstrations, and previous research usually focuses on the data aspect ignoring the model’s effect. In this work, we first revisit the factors contributing to this variance from the model aspect, and find that the demonstration choice is both data- and model-dependent. We further propose a conjecture that *the performance of a demonstration positively correlates with its contribution to the model’s understanding of the test samples*, and accordingly propose a data- and model-dependent demonstration selection method, **TopK + ConE**. Empirically, our method yields consistent improvements in both language understanding and generation tasks with different model scales. Further analyses confirm that, besides the generality and stability under different circumstances, our method provides a unified explanation for the effectiveness of previous methods. Code is publicly available at https://github.com/Romainpkq/revisit_demon_selection_in_ICL.

1 Introduction

Large language models (LLMs, Ouyang et al., 2022; Touvron et al., 2023) have achieved widespread success across many NLP tasks (Zhong et al., 2023; Peng et al., 2023; Lu et al., 2023) due to their remarkable emergent abilities (Wei et al., 2022). One of the most exciting emergent abilities is in-context learning (ICL, Brown et al., 2020b), which utilizes only a few input-output examples to help LLMs make better predictions (Dong et al., 2022). ICL has shown its effectiveness in eliciting LLMs’ advanced capabilities and has (almost) become a common practice in tackling complex tasks.

*Corresponding Authors.

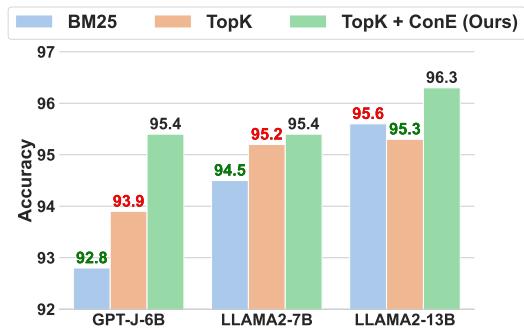


Figure 1: **The different 8-shot performance of data-dependent methods (BM25 and TopK) and Our methods in SST-2.** The colour in the number represents the relative performance between BM25 and TopK. We see that: 1) The data-dependent methods can not obtain optimal demonstrations under different models; 2) Our data- and model-dependent methods can achieve consistent improvement across different models.

However, prior work (Liu et al., 2022; Lu et al., 2022) has found that ICL is very sensitive to the choice of in-context examples and their order in the prompt, and even small changes can result in large variance (Iter et al., 2023).

The sensitivity of ICL motivates researchers to explore methods to identify stable and high-performing demonstrations. Influenced by the success of leveraging a retrieval module to augment neural networks (Hashimoto et al., 2018), the retrieval module has become a standard module in the ICL framework for retrieval demonstrations from a dataset (Liu et al., 2022; Rubin et al., 2022). Extensive research has been conducted to search for demonstrations similar to the test samples (Liu et al., 2022; Su et al., 2023; Robertson et al., 2009). For example, Liu et al. (2022) proposed to select the samples that are closer to the test sample in the embedding space as in-context examples, and Robertson et al. (2009) found that choosing the high word-overlap samples can also improve the ICL performance.

Despite empirical success to some extent, the above methods usually only focus on the test data, overlooking the impact of models. To figure out what factors influence the choice of demonstrations, we revisit the performance of ICL from the model aspect, and accordingly propose a conjecture to understand the effective demonstrations. Specifically, we investigate ICL performance across different retrieval modules and inference models in §2.1. Experimental results show that the ICL performance can largely vary with different models even with the same demonstrations (see Figure 1 as an example), indicating that ***the choice of demonstration is not only dependent on test data but also on the retrieval modules and inference models***. We further propose a corresponding conjecture that ***effective demonstrations are those that enhance the inference model’s understanding of the test input***, and the comparison results between shuffled test input and original test input demonstrate that ***the ICL performance positively correlates with model’s understanding of the test samples***.

Based on the above conjectures, we accordingly propose a demonstration selection method, denoted as **TopK+ConE**. Specifically, we initially employed the TopK (Liu et al., 2022) method to narrow down the pool of demonstration candidates, followed by ranking these candidates based on the **conditional entropy** (estimated by the model itself) of the test sample input. Extensive experiments demonstrate the effectiveness of our method across different model scales. Further analyses show the universality and robustness, and provide a unified view of why previous demonstration selection methods work. Our **contributions** are summarized as follows:

- To the best of our knowledge, we are the first to study the impact of models on the demonstration selection methods. We substantiate that the choice of demonstrations is not only dependent on the test data but also on the retrieval module and inference model.
- We build the connection between ICL performance with the model’s understanding of test inputs. Our findings reveal that ICL performance positively correlates with the model’s understanding of the test samples.
- We propose a data- and model-dependent method **TopK+ConE** to effectively enhance the models’ understanding of test input via

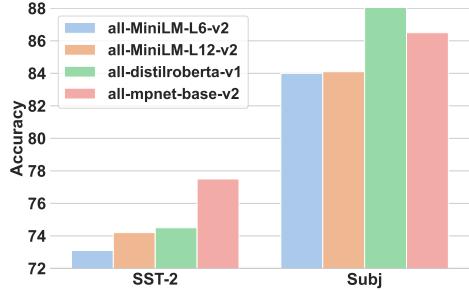


Figure 2: **The 1-shot performance with different retrieval models** on two classification datasets.

reducing the conditional entropy of test input under the inference model.

- We achieve state-of-the-art performance on a series of tasks, and prove the effectiveness and universality of our method. Hopefully, our proposed best practice can be employed by more LLM participants.

2 Revisiting Demonstrations Selection

While in-context learning (ICL, Brown et al., 2020a; Dong et al., 2022) has shown its impressive few-shot performance, recent work has found that LLMs are very sensitive to the selected examples leading to large variances in performance (Zhao et al., 2021). Although many advanced ICL strategies (Robertson et al., 2009; Liu et al., 2022; Wu et al., 2023b) have been proposed to select effective demonstrations, why these demonstrations work and what factors influence their selection have not been fully studied. In this section, we first explore the influencing factors to the demonstration selection and correspondingly propose a conjecture to understand the effective demonstrations.

2.1 Influencing Factors

Preliminaries. The retrieval-based in-context learning paradigm primarily comprises four key components: demonstrations, test samples, the retrieval model and the inference model (Xu et al., 2024a). Previous extensive work (Min et al., 2022; Liu et al., 2022; Su et al., 2023) has found that ICL performance is significantly influenced by the test data, and opting for test-similar demonstrations typically leads to yield superior performance. Although the effect of test data has been widely investigated, the model’s impact has hardly been mentioned. To determine the influence of models, we proceed from both the retrieval model and inference model perspectives.

Impact of Retrieval Models. We first conduct experiments on classification tasks with different retrieval models. Specifically, we conduct experiments on two classification tasks, SST-2 and Subj (Wang et al., 2018), with four sentence-transformer (Reimers and Gurevych, 2019) models, including *all-MiniLM-L6-v2*, *all-MiniLM-L12-v2*, *all-distilroberta-v1* and *all-mpnet-base-v2*. As shown in Figure 2, the performance varies with different retrieval models and different datasets have different best retrievers. We speculate that the variance in model performance primarily arises from distinctions in similarity judgment between the retrieval model and the inference model. A smaller disparity in similarity judgment is expected to result in better in-domain demonstrations, which can improve the ICL performance (Moore and Lewis, 2010; Sia and Duh, 2023).

Impact of Inference Models. The inference model is another factor that may influence the performance of in-context learning. To explore this, we conducted experiments on two classification tasks (e.g., SST-2 and SST-5) employing different inference models in both 1-shot and 3-shot settings. Specifically, we randomly sample different demonstrations 3 times for each test sample and assign them to Random-1, -2, and -3, respectively, and then we assess their performance across various inference models. Results on Figure 3 show that the best demonstration varies across different inference models. For example, the performance of *Random-2* is better than *Random-3* in 1-shot SST-2 setting under *llama2-7b* model, while the situation is totally reversed with *llama2-13b*. We can also notice the same phenomenon under 3-shot settings, which implies increasing the in-context examples can not eliminate the influence of inference models. **Results above show that the choice of demonstrations is model-dependent.**

2.2 Conjecture

Based on the above observations, we find demonstration choice is both data-dependent and model-dependent. Furthermore, Gonen et al. (2023) reveal that the more familiar the model is with prompts, the better the performance of prompts. Inspired by them, we propose a conjecture that *effective demonstrations are those that can help the inference model better understand the test input*.

To verify our assumption, we explore the relationship between the model’s understanding of the

test inputs and ICL performance. We simply employ the straightforward *span shuffle* noise, which first selects sequences consisting of three consecutive tokens, and then randomly change their order, following Ding et al. (2022) to increase the difficulty of test input. Specifically, we first adopt TopK (Liu et al., 2022) method to select the most test-similar demonstrations and compare the ICL performance of noised test samples with their original version. Since the partial word shuffle will not influence people’s reading (Schad and Engbert, 2012; Ward Bowens, 2013), our operation will not largely change the sentence’s meaning. Table 1 lists the results. We can notice that increasing the test samples’ difficulty will lead to a large drop in ICL performance under both 1- and 3-shot settings, which reveals that **ICL performance positively correlates with the model’s understanding of the test samples**.

Method	1-shot			3-shot		
	SST-2	SST-5	Subj	SST-2	SST-5	Subj
Baseline	81.9	38.1	89.8	79.2	39.0	87.6
<i>shuffle</i>	52.7	22.9	54.3	52.6	22.2	55.0
$\bar{\Delta} (\downarrow)$	-29.2	-15.2	-35.5	-26.6	-16.8	-32.6

Table 1: Comparative results of GPT2-XL with origin test input and shuffled test input on several tasks. We observe that the difficulty of test input will largely influence the ICL performance among all these tasks.

3 Method

Based on the above conclusions, we propose a simple and effective data- and model-dependent demonstration selection method, named **TopK + ConE**. Our method is based on the conjecture in section §2.2, which implies effective demonstrations excel in reducing the conditional entropy of the test input under the inference model. It is noteworthy that we compute the conditional entropy of the test input rather than labels. Mathematically, we find the best demonstrations c^* by solving the following optimization problem:

$$c^* = \arg \min_{c \in \mathcal{C}} H_\theta(x|c), \quad (1)$$

where each c represents one possible demonstration group, and $H_\theta(x|c)$ signifies the inference model’s uncertainty regarding the test input x given the demonstrations c , which indicates the degree of the understanding of test input by the inference model.

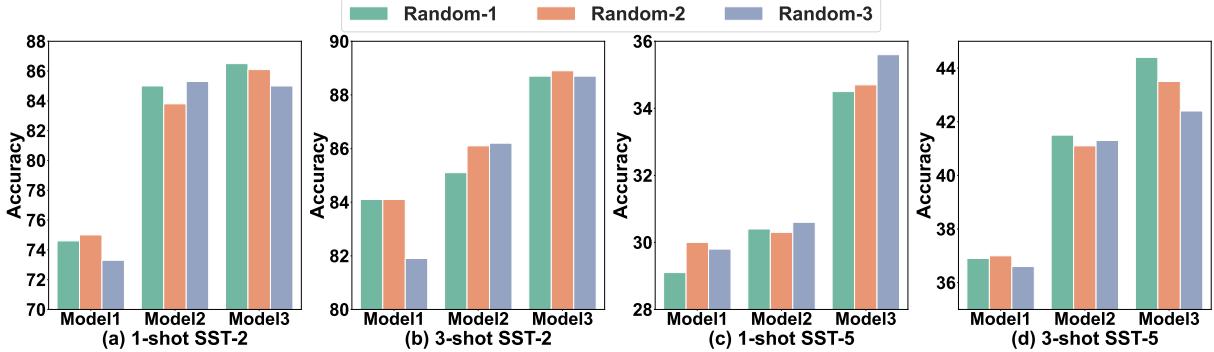


Figure 3: **The performance of different inference models with three randomly sampled demonstrations** for SST-2 and SST-5 datasets. Model1, Model2, Model3 represent GPT-J-6B, LLAMA2-7B, and LLAMA2-13B, respectively. The impact of various demonstrations varies depending on the specific inference models.

The lower the $H_\theta(x|c)$ is, the better the understanding is. The equation (1) can be reformulated as

$$c^* = \arg \min_{c \in C} (H_\theta(x, c) - H_\theta(c)), \quad (2)$$

where $H_\theta(x, c)$ and $H_\theta(c)$ are the cross entropy of the whole prompt (including the demonstrations and test input) and the demonstrations estimated by the inference model, respectively. In other words, we are searching for demonstrations that minimize the difference of the cross-entropy between prompts and demonstrations.

In the practical implementations, considering the huge search space generated by a large number of combinations, enumerating all combinations is infeasible. We adopt the selection-rerank framework proposed in Wu et al. (2023b). Specifically, we first use the selection module to select the candidate demonstrations and then use our method to rank each candidate to get effective demonstrations.

4 Experimental Setup

Models. We perform experiments across different sizes of models, including GPT2-XL (1.5B) (Radford et al., 2019), GPT-j-6b (6B) (Wang and Komatsuzaki, 2021), Llama2-7b (7B) and Llama2-13b (13B) (Touvron et al., 2023), which are decoder-only dense LMs. We also conduct experiments on extensive alignment models, e.g., Llama2-7b-chat and Llama2-13b-chat (Touvron et al., 2023), Vicuna-7b, Vicuna-13b and Deepseek-7b-chat (DeepSeek-AI, 2024) to verify the generalizability of our approach.

Datasets. We conduct a systematic study across 7 natural language understanding (NLU) tasks, including binary, multi-class classification tasks

(SST-2, SST-5 (Socher et al., 2013), CR, Subj (Wang et al., 2018)) and natural language inference tasks: MNLI (Williams et al., 2018) and QNLI (Wang et al., 2018). We also evaluate our method in 4 machine translation tasks, extracted from Flores-200 (Goyal et al., 2022) dataset, which contains 1012 examples for each language.

Baselines. We mainly compare our method with five widely used methods that do not require additional training.

- **Prompting** is a special case of ICL without in-context examples.
- **Random** baseline randomly select in context examples for each testing sample.
- **BM25** (Robertson et al., 2009) baseline uses BM25 to calculate the word-overlap similarity between samples and test input, and select the high similarity samples as demonstrations.
- **TopK** (Liu et al., 2022) baseline uses the nearest neighbors of a given test sample as the corresponding in-context examples.
- **TopK + MDL** (Wu et al., 2023b) adopt a select-then-rank framework, and rank the demonstrations selected by the TopK method based on the Minimum Description Length (MDL) principle.

Evaluation Metrics. We adopt different evaluation methods for different tasks. For classification, we report the performance with the Accuracy. For the translation tasks, we adopt the mostly used language model-based metrics **COMET** (Rei et al., 2020) since they have demonstrated a high correlation with human evaluation and are resilient to

domain shift. Specifically, we use the reference-based metric COMET-20 (*wmt20-COMET-da*) and COMET-22 (*wmt22-COMET-da*) for evaluation, and use the default parameters of "comet-compare" for the significance test¹.

Experimental Details. We use the TopK method to retrieve 30 candidates for each sample, and then rank each candidate using our ConE method. Templates are adopted from Lu et al. (2022); Wu et al. (2023b) and detailed in Table 7. We ran all experiments 3 times with different random seeds and reported the average accuracies. We use 4-shot ICL for GPT2-XL and 8-shot for others, the ablations are in §7. Our codebase is built based on OpenICL (Wu et al., 2023a).

5 Main Results

5.1 Natural Language Understanding Tasks

We first verify the effectiveness of our method in NLU Tasks. Specifically, we conduct experiments on 7 classification tasks, including binary classification tasks, multi-class classification tasks, and natural language inference tasks. Based on the results on Table 2 and Figure 4, we can find that:

Our method brings consistent performance improvements on almost all types of tasks. Results in Table 2 show the superior performance of our approach compared to the existing state-of-the-art method, *TopK+MDL*, across the majority of tasks, resulting in an average accuracy improvement of 1.2%. Compared with our selection method *TopK*, our method considerably improves the performance on 6 tasks out of the total 7 tasks, yielding an average gain of 1.8%, proving the effectiveness of improving the model’s understanding to test input. Furthermore, it is noteworthy that our approach can achieve significant improvements in challenging tasks, such as the Subj and QNLI tasks, respectively bringing 4.6% and 1.9% gains compared to the previously optimal methods, demonstrating the superior performance of our method for hard-to-understanding tasks.

Our method brings gains across different model sizes. Figure 4 presents the average performance across 7 Natural Language Understanding (NLU) tasks using various inference models, ranging in size from 1.5B (*GPT2-XL*) to 13B (*Llama2-13B*). Results reveal that advanced ICL methods usually

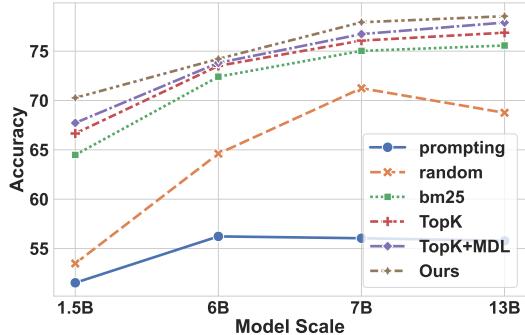


Figure 4: **The average performance of 7 NLU tasks across different model scales.** Our method consistently outperforms previous methods across model scales.

can achieve better performance when we scale up the model size, while prompting and random ICL methods will produce unstable results. Notably, our approach consistently outperforms previous methods across different model scales, particularly in the case of GPT2-XL, which yields an average gain of 2.6% and 3.6% compared to *TopK+MDL* and *TopK* methods.

5.2 Natural Language Generation Tasks

We further evaluate our method on NLG tasks, i.e. machine translation. Recent study (Hendy et al., 2023) reveals that LLMs have achieved comparable or better performance on par with their best-supervised counterpart systems (Zan et al., 2022) in competing WMT Chinese-English tasks. We conduct experiments in 4 language pairs, including English-centric language pairs and non-English-centric language pairs.

Results. The results across different language pairs under different settings are presented in Table 3. Obviously, our method can consistently improve the performance of ICL in terms of COMET score compared with TopK in both English-centric and non-English-centric language pairs. Especially in non-English-centric language pairs, our method brings +1.1 and +2.2 COMET20 score improvement in Ru⇒De and De⇒Ru under the 3-shot setting, respectively. We attribute this to the improvement of the model’s understanding of the test sample, and the more difficult the sample, the greater the benefit from our method. Furthermore, we can notice that previous advanced ICL methods do not always work, especially for non-English centric language pairs, while our method can consistently achieve the best performance under the 3-shot settings, demonstrating the effectiveness of

¹<https://github.com/Unbabel/COMET>

Method	SST-2	CR	Subj	SST-5	AGNews	MNLI	QNLI	Average
Prompting	68.7	81.1	49.4	25.3	67.0	47.5	53.3	56.0 (+21.9)
Random	94.4	92.3	70.9	50.4	83.5	51.0	56.2	71.2 (+6.8)
BM25	94.5	92.8	76.8	52.6	92.5	57.0	59.0	75.0 (+2.9)
TopK	95.2	92.8	80.4	52.6	92.4	57.8	61.3	76.1 (+1.8)
TopK + MDL	95.1	93.4	81.2	52.7	92.3	57.9	64.5	76.7 (+1.2)
Ours	95.4	93.1	85.8	52.5	92.8	59.5	66.4	77.9

Table 2: **Performance of different methods across 7 Natural Language Understanding (NLU) tasks** on Llama2-7B model. The best results are in **bold**. We can see that our method improves the performance of almost all task types. Numbers in the parenthesis represent the relative improvements our method achieved over baselines.

Method	En⇒Zh		Zh⇒En		Ru⇒De		De⇒Ru	
	COMET20	COMET22	COMET20	COMET22	COMET20	COMET22	COMET20	COMET22
-w/ 1-shot								
Random	35.7	81.5	60.9	85.1	44.0	79.8	52.4	83.6
BM25	35.1	81.3	60.9	85.1	42.2	79.5	50.2	83.4
TopK	35.9	81.5	61.0	85.1	43.9	79.7	49.7	83.3
Ours	37.1[†]	81.7[†]	61.7[†]	85.4[†]	43.9	79.9	51.8 [†]	83.8[†]
-w/ 3-shot								
Random	40.1	82.4	62.7	85.5	47.8	80.6	54.6	84.0
BM25	39.6	82.3	62.3	85.4	47.0	80.5	53.2	83.9
TopK	39.9	82.4	63.3	85.6	46.8	80.4	53.1	83.9
Ours	40.7[†]	82.6[†]	63.3	85.7	47.9[†]	80.8[†]	55.3[†]	84.5[†]

Table 3: **Performance on different methods across 4 language pairs on Llama2-7b**. The best results are in **bold**. “[†]” indicates a statistically significant difference from the TopK baseline ($p < 0.05$).

our method on generation tasks.

6 Analysis

To further demonstrate the effectiveness and generality of our method, we conduct further analyses on NLU tasks (with the GPT2-XL model) and NLG tasks (with Llama2-7b).

Our ConE method is complementary to previous approaches. To further explore the generality of our method, we combine ConE with different selection methods, e.g. random and BM25, in binary and multi-choice classification tasks. The results in Figure 6 (a, b) show that ConE can further significantly improve the baseline performance in different types of tasks. Especially in SST-2 tasks with the Random method, ConE brings +7.5 score improvement, which indicates that ConE is complementary to previous approaches and can further improve their performance. We can also notice that TopK + ConE achieves better performance compared with other methods, hence we choose TopK as our selection method because of its simplicity and effectiveness.

Our method works for mix-domain demonstration pools. Previous results have shown the superior performance of our method in single-domain demonstration pools. Now, we evaluate the effectiveness of our method in mixed demonstration pools, which have demonstrations from different domains. Specifically, we evaluate the performance of our method in three domains, e.g., e-commerce, news and social, with a mix-domain demonstrations pool in WMT22 translation task². Experimental results in Table 4 show that our method can achieve consistent improvements in three domains with 3-shot ICL, especially in Zh⇒En, which achieve over 1.0 COMET improvement across three domains, showing that *our method also works for mix-domain demonstration pools*.

Our method works for aligned chat models. To verify the effectiveness of our method for the chat LLMs, we conducted extensive experiments on different instruction-tuned and RLHF-tuned LLMs, including Vicuna, LLaMA-chat, and DeepSeek-chat. The results in Figure 5 show that our method

²<https://www.statmt.org/wmt22/translation-task.html>

Method	Zh⇒En			En⇒Zh		
	ecommerce	news	social	ecommerce	news	social
-w/ 1-shot						
random	3.7	29.8	31.2	33.0	17.8	6.8
TopK	6.4	30.6	32.4	32.6	18.1	6.1
Ours	6.0	33.2[†]	32.4[†]	36.1[†]	21.0[†]	4.8
-w/ 3-shot						
random	8.1	33.3	33.4	34.3	22.4	11.3
TopK	7.5	35.3	33.3	36.7	24.1	11.9
Ours	9.5[†]	36.3[†]	34.4[†]	37.0[†]	25.2[†]	12.5[†]

Table 4: **Performance of our method for domain dataset with a mixed-domain demonstration pool** with inference model Llama2-7b. “ \dagger ” indicates a statistically significant difference from the TopK baseline ($p < 0.05$).

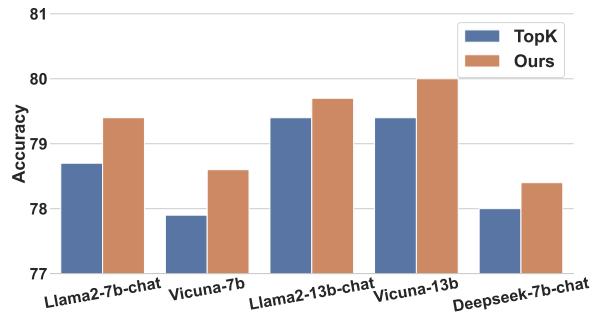


Figure 5: **The average performance of different chat models** in 7 NLU tasks.

can achieve consistent improvement in different models, demonstrating that *our method also works for instruction-tuned and safety-enhanced models*.

7 Impact of hyperparameter

In this section, we conduct ablation studies on the hyperparameters in our method.

Impact of In-context Examples. We gradually increase the number of in-context examples (denoted as N) from 0 (prompting) to 16. The results are listed in Figure 7(a, b), we see that increasing N usually can consistently improve the performance on average, but when $N=16$ the ICL performance in GPT2-XL degrades. Through further analysis, we found that the decrease comes from the constraint of the maximum sentence length of the model (GPT2-XL), and the phenomenon even occurs when we set N as 8 for GPT2-XL. Hence, we choose $N=4$ for GPT2-XL, and $N=8$ for other models. Note that our method can consistently outperform the TopK method, and increasing the in-context examples can further improve our method.

Impact of Candidate Numbers. As mentioned above, our method comprises two modules: the

TopK selection and the ConE reranking. The selection module will reduce the space of in-context examples to speed up the whole process. Hence we explore the impact of the candidate numbers selected by TopK. The results in Figure 7(c) list the performance of 4 in-context examples with the GPT2-XL model. We can notice that our method is always better than the baseline TopK, and increasing the number of candidates can further improve the performance. Based on the results, we set the default candidate number as 30.

8 Discussion

Whether our method can partially explain why previous ICL methods work? Intuitively, enhancing the model’s understanding to test input is one of the reasons why previous methods work. To prove this, we calculate the conditional entropy of the test input with respect to previous baselines across three classification tasks. The results presented in Figure 6(c) show that the previous methods will also reduce the conditional entropy of test samples in all three tasks, which demonstrate that previous ICL methods can also be explained by our conjecture. These results show the universality of our conjecture.

Whether our method is sensitive to the demonstration order? Previous studies have proven that ICL is very sensitive to the order of in-context examples (Lu et al., 2022). To explore the sensitivity of our methods for the order of in-context examples, we randomize the order of our chosen demonstrations on three classification tasks and compare the stability with Random and TopK methods. Results on Table 5 show that our method can achieve better average performance with smaller variance among all tasks, demonstrating that *our method could alleviate the order sensitivity issue in the ICL framework*.

9 Related Work

Despite that large language models have shown their surprising zero-shot performance in various tasks, even including complex reason/ agent tasks (Wang et al., 2024b; Ren et al., 2024; Zhang et al., 2024; Zhong et al., 2024). Recent works show that ICL can effectively elicit their capability and further improve LLMs’ performance (Dong et al., 2022). Besides effectiveness, ICL may provide format guidance to alleviate the prompt bias during language model inference (Xu et al., 2024b).

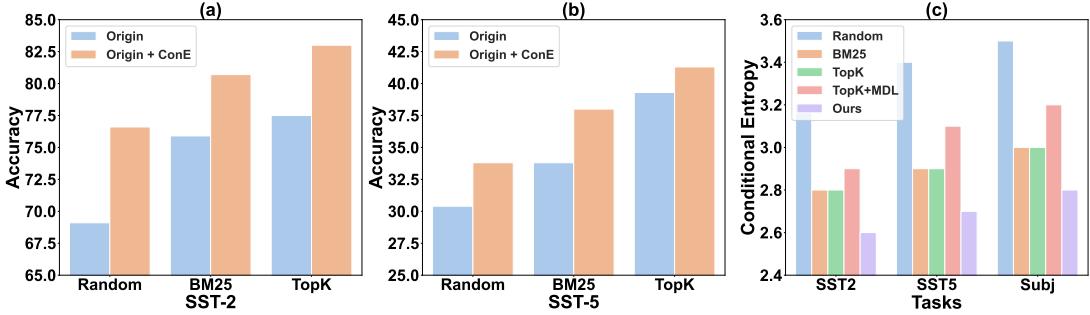


Figure 6: (a, b) **The effect of our method with different selection methods** in SST-2 and SST-5, **origin** represents the baseline method without our ConE method, while **Origin + ConE** signifies with our ConE method. (c) **The conditional entropy of the test input** with different ICL methods.

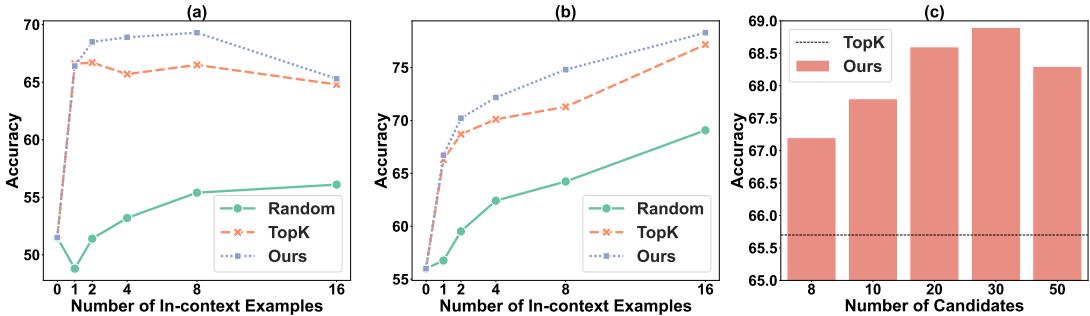


Figure 7: **The average performance of ablation experiments.** (a, b) Impact of the number of in-context examples for GPT2-XL and Llama2-7b; (c) Impact of the number of candidates selected by the TopK method.

Method	SST-2		Subj		CR	
	Avg.	Var.	Avg.	Var.	Avg.	Var.
Random	68.8	0.90	56.7	0.49	67.8	4.00
TopK	78.6	0.56	86.2	0.20	73.9	0.61
Ours	82.0	0.26	91.0	0.05	81.0	0.26

Table 5: **The average performance and variance of 10 random permutations of four in-context examples** for GPT2-XL. The best results are in **bold**. *Our method achieves consistently better average performance with lower variance.*

However, the performance of ICL is unstable (Lu et al., 2022), and the small change of in-context examples and their order can result in a large variance in performance.

Motivated by the instability of the ICL performance, in-context example selection methods have been widely investigated. Lu et al. (2022) first propose a validation-free corpus-level method for determining the optimal order of in-context examples. However, they only investigate the influence of order without proposing how to better select in-context examples. Inspired by the success of retrieval modules in augmenting neural networks,

Liu et al. (2022) find examples that are close to each test sample in embedding space that can serve as a good choice for ICL. Following the finding of Liu et al. (2022), Su et al. (2023) subsequently extended their method by incorporating increased diversity in the selection of in-context examples. However, why these methods work is still unclear and the methods only consider the influence from the data aspect.

Unlike the data-dependent demonstration selection methods, model-dependent methods are rarely explored. Wu et al. (2023b) proposed a demonstration rank method grounded in the minimum description length principle, which utilizes the inference model to select the optimal in-context example organization. However, their ranked in-context organizations are randomly sampled, which may limit their performance. Wang et al. (2024a) proposed a novel framework to iteratively train dense retrievers to identify high-quality in-context examples for LLMs. However, they need additional training, which is costly for practitioners. Furthermore, both methods neglected to investigate whether and how the inference model affects ICL performance.

On the other hand, although some previous methods (Wu et al., 2023b; Iter et al., 2023; Wang et al.,

2023) have emphasized the significance of understanding the test samples, their primary emphasis lies in the confidence of test labels, neglecting that of test input. For instance, Wu et al. (2023b) searches the demonstrations capable of losslessly compressing testing labels, and Iter et al. (2023) identify the in-domain demonstrations through the cross-entropy difference of test labels computed by the small model fine-tuned in demonstrations. While Wang et al. (2023) propose to reweight label anchors to improve ICL performance. Gonen et al. (2023) found that using perplexity could be a good heuristic for prompt selection, while the effect for ICL has not been investigated.

10 Conclusion

In this paper, we take the first step to investigate the factors that influence the choice of demonstrations in ICL from the model perspective, and find that the demonstration selection is both data- and model-dependent. Based on the findings, we conjecture that effective demonstrations can improve the inference model’s understanding to test input, and correspondingly propose a data- and model-dependent selection method. Empirical results suggest that our method can significantly outperform the previous ICL method. Further analysis confirms the generalization of our method and our approach can provide a unified explanation for previous studies.

Limitations

Our work has several potential limitations. First, given the limited computational budget, we only validate our **TopK + ConE** on the 1.5B-13B LLMs. It will make our work more convincing if scaling the experiments up to the larger model size, e.g., 70B. On the other hand, our method introduces some computational budgets during the inference for select demonstrations, which may be unacceptable for extremely large LLMs. It is meaningful to explore a more efficient method to measure the model’s understanding to test input to accelerate the process of demonstration selection, which is in our future work.

Ethic Statements

We take ethical considerations very seriously and strictly adhere to the ACL Ethics Policy. This paper focuses on the in-context learning behaviour of LLMs and proposes a data- and model-dependent demonstration selection method to improve ICL

performance. To explore the influencing factors of ICL, we revisit the demonstration selection Strategies from model aspect, and propose a conjecture to find effective demonstrations. However, it should be noted that all pretrained models and evaluation datasets used in this study are publicly available and have been widely adopted by researchers. We do not proactively introduce additional data or models that may cause ethical issues, and we believe that our proposed method will help alleviate ethical issues.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62377002). Xuebo Liu was sponsored by CCF-Tencent Rhino-Bird Open Research Fund. We are grateful to the anonymous reviewers and the area chair for their insightful comments and suggestions.

References

- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020a. *Language models are few-shot learners*. *NeurIPS*.
- 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. 2020b. *Language models are few-shot learners*. In *NeurIPS*.
- DeepSeek-AI. 2024. *Deepseek llm: Scaling open-source language models with longtermism*. *arXiv preprint arXiv:2401.02954*.
- Liang Ding, Kefin Peng, and Dacheng Tao. 2022. *Improving neural machine translation by denoising training*. *arXiv preprint*.
- Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2022. *A survey for in-context learning*. *arXiv preprint*.
- Hila Gonen, Srini Iyer, Terra Blevins, Noah Smith, and Luke Zettlemoyer. 2023. *Demystifying prompts in language models via perplexity estimation*. In *Findings of EMNLP*.

- Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. **The Flores-101 evaluation benchmark for low-resource and multilingual machine translation.** *TACL*.
- Tatsunori B Hashimoto, Kelvin Guu, Yonatan Oren, and Percy S Liang. 2018. **A retrieve-and-edit framework for predicting structured outputs.** *NeurIPS*.
- Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. **How good are gpt models at machine translation? a comprehensive evaluation.** *arXiv preprint*.
- Dan Iter, Reid Pryzant, Ruochen Xu, Shuohang Wang, Yang Liu, Yichong Xu, and Chenguang Zhu. 2023. **In-context demonstration selection with cross entropy difference.** In *EMNLP*.
- Jiachang Liu, Dinghan Shen, Yizhe Zhang, William B Dolan, Lawrence Carin, and Weizhu Chen. 2022. **What makes good in-context examples for gpt-3?** In *DeeLIO*.
- Qingyu Lu, Baopu Qiu, Liang Ding, Kanjian Zhang, Tom Kocmi, and Dacheng Tao. 2023. **Error analysis prompting enables human-like translation evaluation in large language models: A case study on chatgpt.** *arXiv preprint*.
- 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 *ACL*.
- Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. **Rethinking the role of demonstrations: What makes in-context learning work?** In *EMNLP*.
- Robert C. Moore and William Lewis. 2010. **Intelligent selection of language model training data.** In *ACL*.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. **Training language models to follow instructions with human feedback.** *NeurIPS*.
- Keqin Peng, Liang Ding, Qihuang Zhong, Li Shen, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2023. **Towards making the most of chatgpt for machine translation.** In *Findings of EMNLP*.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. **Language models are unsupervised multitask learners.** *OpenAI blog*.
- Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. **COMET: A neural framework for MT evaluation.** In *EMNLP*.
- Nils Reimers and Iryna Gurevych. 2019. **Sentence-BERT: Sentence embeddings using Siamese BERT-networks.** In *EMNLP*.
- Zhiyao Ren, Yibing Zhan, Baosheng Yu, Liang Ding, and Dacheng Tao. 2024. **Healthcare copilot: Eliciting the power of general llms for medical consultation.** *arXiv preprint*.
- Stephen Robertson, Hugo Zaragoza, et al. 2009. **The probabilistic relevance framework: Bm25 and beyond.** *Foundations and Trends® in Information Retrieval*.
- Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. **Learning to retrieve prompts for in-context learning.** In *NAACL*.
- Daniel J Schad and Ralf Engbert. 2012. **The zoom lens of attention: Simulating shuffled versus normal text reading using the swift model.** *Visual Cognition*.
- Suzanna Sia and Kevin Duh. 2023. **In-context learning as maintaining coherency: A study of on-the-fly machine translation using large language models.** In *MTSummit*.
- Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. **Recursive deep models for semantic compositionality over a sentiment treebank.** In *EMNLP*.
- Hongjin Su, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. 2023. **Selective annotation makes language models better few-shot learners.** In *ICLR*.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. **Llama 2: Open foundation and fine-tuned chat models.** *arXiv preprint*.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. **GLUE: A multi-task benchmark and analysis platform for natural language understanding.** In *EMNLP*.
- Ben Wang and Aran Komatsuzaki. 2021. **GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model.** <https://github.com/kingoflolz/mesh-transformer-jax>.
- 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 *EMNLP*.
- Liang Wang, Nan Yang, and Furu Wei. 2024a. **Learning to retrieve in-context examples for large language models.** In *EACL*.
- Shuai Wang, Liang Ding, Li Shen, Yong Luo, Bo Du, and Dacheng Tao. 2024b. **Oop: Object-oriented programming evaluation benchmark for large language models.** *arXiv preprint*.

Saundra Ward Bowens. 2013. *The relationship between using the scrambled words reading strategy and the vocabulary of struggling readers*. *ProQuest LLC*.

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*. *TMLR*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. *A broad-coverage challenge corpus for sentence understanding through inference*. In *NAACL*.

Zhenyu Wu, Yaxiang Wang, Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Jingjing Xu, and Yu Qiao. 2023a. *OpenICL: An open-source framework for in-context learning*. In *ACL*.

Zhiyong Wu, Yaxiang Wang, Jiacheng Ye, and Lingpeng Kong. 2023b. *Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering*. In *ACL*.

Xin Xu, Yue Liu, Panupong Pasupat, Mehran Kazemi, et al. 2024a. *In-context learning with retrieved demonstrations for language models: A survey*. *arXiv preprint*.

Ziyang Xu, Keqin Peng, Liang Ding, Dacheng Tao, and Xiliang Lu. 2024b. *Take care of your prompt bias! investigating and mitigating prompt bias in factual knowledge extraction*. In *LREC-COLING*.

Changtong Zan, Keqin Peng, Liang Ding, Baopu Qiu, Boan Liu, Shuai He, Qingyu Lu, Zheng Zhang, Chuang Liu, Weifeng Liu, Yibing Zhan, and Dacheng Tao. 2022. *Vega-MT: The JD explore academy machine translation system for WMT22*. In *WMT*.

Yuqi Zhang, Liang Ding, Lefei Zhang, and Dacheng Tao. 2024. *Intention analysis makes llms a good jailbreak defender*. *arXiv preprint*.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. *Calibrate before use: Improving few-shot performance of language models*. In *ICML*.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2023. *Can chatgpt understand too? a comparative study on chatgpt and fine-tuned bert*. *arXiv preprint*.

Qihuang Zhong, Kang Wang, Ziyang Xu, Juhua Liu, Liang Ding, Bo Du, and Dacheng Tao. 2024. *Achieving >97% on gsm8k: Deeply understanding the problems makes llms better reasoners*. *arXiv preprint*.

A Datasets

Natural Language Understanding (NLU) Dataset information is detailed in Table 6. All NLU datasets are loaded from HuggingFace Hub. For

most NLU datasets, we report the results on the test set; while for the datasets MNLI and QNLI, we report the results on the validation set due to restricted access to their test sets.

B Templates

The templates of NLU tasks used in this paper are detailed in Table 7. For NLG tasks, we adopted templates as [src]: <X’> [tgt]: <Y’>, where [src] and [tgt] represent the source and target language names of the test language pair, respectively, and placeholders <X’> and <Y’> will be replaced by source and target sentences.

Dataset	Task	# of Classes	Data Split
SST-2	Sentiment Classification	2	6920/872/1821
SST-5	Sentiment Classification	5	8544/1101/2210
CR	Sentiment Classification	2	3394/0/376
Subj	Subjectivity Analysis	2	8000/0/2000
AgNews	Topic Classification	4	120000/0/7600
MNLI	Natural Language Inference	3	392702/19647/19643
QNLI	Natural Language Inference	2	104743/5463/5463

Table 6: **Details of NLU datasets.**

Task	Prompt	Class
SST-2	Review: "<X>" Sentiment: positive	positive
	Review: "<X>" Sentiment: negative	negative
SST-5	Review: "<X>" Sentiment: terrible	terrible
	Review: "<X>" Sentiment: bad	bad
	Review: "<X>" Sentiment: okay	okay
	Review: "<X>" Sentiment: good	good
	Review: "<X>" Sentiment: great	great
Subj	Input: "<X>" Type: objective	objective
	Input: "<X>" Type: subjective	subjective
CR	Review: "<X>" Sentiment: positive	positive
	Review: "<X>" Sentiment: negative	negative
AgNews	"<X>" It is about world.	World
	"<X>" It is about sports.	Sports
	"<X>" It is about business.	Business
	"<X>" It is about science and technology.	Sci/Tech
MNLI	<C> Can we know <X>? Yes.	Entailment
	<C> Can we know <X>? Maybe.	Neutral
	<C> Can we know <X>? No.	Contradiction
QNLI	<C> Can we know <X>? Yes.	Entailment
	<C> Can we know <X>? No.	Contradiction

Table 7: **Templates of NLU tasks.** Placeholders (e.g., <X> and <C>) will be replaced by real inputs.