Comparative Analysis of Demonstration Selection Algorithms for In-Context Learning in Large Language Models (Student Abstract)

Dong Shu[1], **Mengnan Du**[2]
[1]Northwestern University
[2]New Jersey Institute of Technology
dongshu2024@u.northwestern.edu, mengnan.du@njit.edu

Abstract

Demonstration selection algorithms play a crucial role in optimizing Large Language Models' (LLMs) in-context learning performance. Despite numerous proposed algorithms, their comparative effectiveness remains understudied. We present a comprehensive evaluation of six state-of-the-art demonstration selection algorithms across five datasets, examining both their effectiveness and computational efficiency. Our findings reveal significant trade-offs: while some demonstration selection algorithms achieve superior accuracy, they incur substantial computational costs. We also discover that increasing demonstration examples doesn't consistently improve performance, and some sophisticated algorithms struggle to outperform random selection in certain scenarios. These insights provide valuable benchmarks for future algorithm development and practical implementation. Our code is available at https://github.com/Tizzzzy/Demonstration_Selection_Overview.

Introduction

Large Language Models (LLMs) excel in in-context learning, where they adapt to new tasks using few-shot learning without requiring additional training (Xie et al. 2021). However, their performance is highly sensitive to the quality of the provided demonstrations. Recently, various demonstration selection algorithms have been developed to enhance this quality by selecting the most informative and relevant examples from the data pool. These algorithms have significantly reduced the time required for LLMs to address unseen tasks and have greatly improved their overall performance.

Despite the success of these approaches, the effectiveness of selected examples and the efficiency of the selection and inference processes are not well understood. This lack of understanding makes it challenging for future research to identify areas for improvement and makes it difficult to use these algorithms in real-life situations. This paper aims to evaluate these two critical aspects, offering insights that will serve as a benchmark for future research.

Demonstration Selection Algorithms

In this paper, we aim to provide a comprehensive analysis of current demonstration selection methods and their impact on LLM performance. In this study, we compare six demonstration selection algorithms for in-context learning: Concept Based Demonstration Selection (CBDS) (Wang et al. 2024), Rethinking Demonstrations direct (RD-direct) and channel (RD-channel) (Min et al. 2022), LLM Retriever (Wang, Yang, and Wei 2023), UPRISE (Cheng et al. 2023), and OpenICL TopK (Wu et al. 2023). These algorithms employ various strategies, from leveraging latent concepts to using retrieval models, to select the best examples for LLMs. Additionally, we include OpenICL Random as a baseline, which randomly selects examples from the data pool.

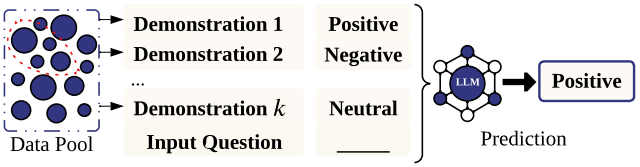

Figure 1: An overview of demonstration selection algorithms: These algorithms select demonstrations from the data pool, which the LLMs then use to generate answers.

Experiments

Experimental Settings

Datasets: The demonstration selection algorithms will be evaluated on 5 datasets, categorized into two groups: classification and multi-choice. The classification datasets include GLUE-MRPC, GLUE-QNLI, and GLUE-SST2 (Wang et al. 2018). The multi-choice datasets are CMSQA (Talmor et al. 2018) and SWAG (Zellers et al. 2018).

Metrics: Given that all datasets are either classification or multi-choice, accuracy is an appropriate metric for evaluating the performance. To show the algorithm's computational efficiency, we present the results in seconds.

Implementation Details: All of the demonstration selection algorithms were tested on LLaMa3-8B (Dubey et al. 2024) to evaluate their effectiveness. As shown in Figure 1, each algorithm first selects k demonstration examples from the available data pool based on the input question, which serve as in-context learning exemplars. These exemplars, along with the input question, are then fed into LLaMa3 for

Datasets	MRPC				QNLI				CMSQA				SWAG				SST2			
	k=4	k=8	k=10	k=20	k=4	k=8	k=10	k=20	k=4	k=8	k=10	k=20	k=4	k=8	k=10	k=20	k=4	k=8	k=10	k=20
CBDS	62.3	**80.8**	**78.8**	57.3	51.1	50.7	54.4	54.8	29.2	29.4	34.4	43.2	36.8	38.1	34.6	36.9	91.4	**93.9**	93.5	73.2
RD-direct	40.6	40.6	34.2	34.0	33.6	35.1	35.9	35.5	42.3	42.8	43.1	41.2	56.2	56.6	57.3	56.4	91.9	75.7	**94.6**	83.8
RD-channel	44.0	45.1	45.3	42.9	46.7	47.8	44.9	38.8	53.6	53.4	56.6	48.8	53.2	52.0	51.8	57.8	91.2	87.9	79.3	76.4
LLM Retriever	74.7	75.4	75.9	**76.2**	71.5	72.3	72.9	72.7	50.6	52.2	52.8	53.8	65.1	64.7	64.9	**64.7**	**93.7**	93.8	93.6	93.5
UPRISE	**75.2**	74.7	75.9	76.0	**81.8**	**81.9**	**81.8**	**81.1**	**61.4**	**63.2**	**64.4**	**66.2**	**69.0**	**66.8**	**69.9**	61.9	93.4	92.5	92.4	91.3
OpenICL TopK	64.1	64.2	65.0	65.2	78.1	74.7	70.8	69.9	34.2	37.6	36.7	32.9	35.4	36.7	37.1	34.9	91.2	92.1	92.3	**94.5**
Random Selection	59.0	62.6	60.1	62.3	51.8	48.3	47.9	47.4	26.7	28.1	28.1	28.5	29.6	31.8	31.4	31.1	86.2	89.5	91.3	91.7

Table 1: Effectiveness of the Demonstration Selection Algorithms

Datasets	MRPC		QNLI	
	selection	inference	selection	inference
CBDS	5.44	1.82	5.35	1.80
RD-direct	$2.34 * 10^{-3}$	1.80	$2.37 * 10^{-3}$	1.81
RD-channel	$2.24 * 10^{-3}$	1.80	$2.50 * 10^{-3}$	1.79
LLM Retriever	$8.32 * 10^{-1}$	12.81	$6.74 * 10^{-1}$	12.84
UPRISE	3.06	3.67	3.23	3.66
OpenICL TopK	$1.46 * 10^{-1}$	1.20	$2.22 * 10^{-1}$	1.19
Random Selection	$5.24 * 10^{-6}$	1.11	$5.63 * 10^{-6}$	1.10

Table 2: Efficiency of Demonstration Selection Algorithms

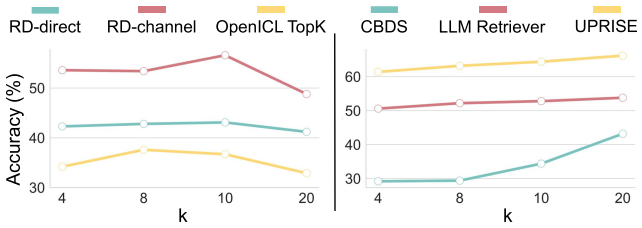

Figure 2: A visualize trend of the algorithms effectiveness

evaluation. We measured the absolute time each algorithm took to select k demonstration examples per single input, as well as the model inference time for the same input. To ensure fairness, we excluded all offline computational time, such as pre-training or other preliminary computations, for all algorithms. The values of k considered in this study were $\{4, 8, 10, 20\}$. All experiments were conducted using an NVIDIA RTX A6000 GPU, with the LLaMa3-8B model directly loaded from Huggingface, and we used the default sampling parameters such as temperature and top_p.

Main Results

Table 1 presents the effectiveness of each algorithm across different datasets and varying numbers of k in-context examples. The highest accuracy in each column is highlighted in bold format. We evaluated the computational efficiency of the six algorithms and a baseline, and report the results in Table 2. Our key takeaways are listed as follows:

- Contrary to expectations, not all demonstration selection algorithms consistently outperform random selection. While some algorithms show significant improvements, others struggle to surpass the baseline set by random selection in certain scenarios. This finding challenges the assumption that sophisticated selection methods always yield better results and highlights the importance of thorough evaluation before implementation.
- Among the compared algorithms, CBDS and UPRISE stand out for their high accuracy but also for their poor efficiency. On average, these algorithms require over 5 seconds and 3 seconds respectively per sample for demonstration selection. Referring to Table 1, out of 20 combinations (4 k values × 5 datasets), CBDS or UPRISE achieved the best accuracy in 14 cases. However, this superior accuracy comes at the cost of high computational complexity. The low efficiency of these algorithms makes them challenging to apply in real-world applications where quick response times are crucial.
- The RD algorithm provides both direct and channel approaches for presenting demonstration examples to the model. In our comparisons, we used "demonstrations with gold labels" for both the direct and channel approaches. We found that the channel approach generally outperforms the direct approach. This observation aligns with the results presented in Figures 8-10 of the original paper (Min et al. 2022). A possible explanation is that the channel approach better aligns the input features with the class labels, making it easier for the model to understand.
- Our analysis reveals that increasing the number of demonstrations (k) does not always lead to better performance. The relationship between k and accuracy is not monotonic and varies significantly across tasks and algorithms. In Figure 2, we visualize the effectiveness trends of six algorithms on the CMSQA dataset. We observed two primary patterns: (1) as shown on the left, performance peaks at a certain k before declining, and (2) on the right, accuracy continues to improve as k increases. Thus, it is crucial to tune the number of demonstrations for each specific task and algorithm combination.

Conclusions and Future Work

In this paper, our comprehensive evaluation of demonstration selection algorithms reveals significant variations in both effectiveness and efficiency across different tasks and datasets. While some algorithms show promising results, the trade-offs between accuracy and computational cost present challenges for real-world applications. Our findings highlight the need for task-specific optimization and suggest potential avenues for future research. These include developing adaptive algorithms to adjust demonstration numbers by task complexity, using advanced retrieval methods, and balancing effectiveness with efficiency.

Acknowledgments

The work is in part supported by NSF #2310261. The views and conclusions in this paper are those of the authors and should not be interpreted as representing funding agencies.

References

Cheng, D.; Huang, S.; Bi, J.; Zhan, Y.; Liu, J.; Wang, Y.; Sun, H.; Wei, F.; Deng, D.; and Zhang, Q. 2023. Uprise: Universal prompt retrieval for improving zero-shot evaluation. *arXiv preprint arXiv:2303.08518*.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Min, S.; Lyu, X.; Holtzman, A.; Artetxe, M.; Lewis, M.; Hajishirzi, H.; and Zettlemoyer, L. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Talmor, A.; Herzig, J.; Lourie, N.; and Berant, J. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Wang, L.; Yang, N.; and Wei, F. 2023. Learning to retrieve in-context examples for large language models. *arXiv preprint arXiv:2307.07164*.

Wang, X.; Zhu, W.; Saxon, M.; Steyvers, M.; and Wang, W. Y. 2024. Large language models are latent variable models: Explaining and finding good demonstrations for incontext learning. *Advances in Neural Information Processing Systems*, 36.

Wu, Z.; Wang, Y.; Ye, J.; Feng, J.; Xu, J.; Qiao, Y.; and Wu, Z. 2023. Openicl: An open-source framework for in-context learning. *arXiv preprint arXiv:2303.02913*.

Xie, S. M.; Raghunathan, A.; Liang, P.; and Ma, T. 2021. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*.

Zellers, R.; Bisk, Y.; Schwartz, R.; and Choi, Y. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *arXiv preprint arXiv:1808.05326*.