MDR: Model-Specific Demonstration Retrieval at Inference Time for In-Context Learning

Huazheng Wang*, **Jinming Wu***, **Haifeng Sun**†, **Zixuan Xia**, **Daixuan Cheng**,
Jingyu Wang, **Qi Qi**†, **Jianxin Liao**
State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications
{wanghz,wjm_18,hfsun,zjjs2019xzx,wangjingyu,qiqi8266}@bupt.edu.cn
{daixuancheng6,jxlbupt}@gmail.com

Abstract

Recently, retrieval-based in-context learning (ICL) methods for selecting demonstrations have been widely investigated. Existing methods train a dense retriever to retrieve the most appropriate demonstrations for a given test query, which improves ICL performance. However, we find that distinct LLMs exhibit different biases for "what is a good demonstration" since they possess differences in training data, model architectures and training methods. As a result, a demonstration suitable for one LLM may not be appropriate for others. Previous approaches ignore the model bias and fail to retrieve the most appropriate demonstrations for different inference LLMs, resulting in a degradation of ICL performance. To address this problem, we propose a simple yet effective metric to evaluate the appropriateness of demonstrations for a specific inference LLM. Furthermore, we introduce a **M**odel-specific **D**emonstration **R**etrieval (MDR) method for ICL at inference time, which considers the biases of different LLMs. We test MDR on seen and unseen tasks with multi-scale inference LLMs, such as `GPT-Neo-2.7B`, `LLaMA-7B` and `Vicuna-13B`. Experiments on 23 datasets across 11 data domains highlight the remarkable effectiveness of MDR, showcasing improvements of up to 41.2% in comparison to methods that neglect model biases. Our code will be publicly available at: https://github.com/kiming-ng/MDR.

1 Introduction

Large Language Models (LLMs) such as `GPT-3` (Brown et al., 2020), have emerged impressive abilities of handling a wide range of tasks. In-context Learning (ICL) (Brown et al., 2020), a new learning paradigm, allows LLMs to perform multi-tasks by observing a few demonstrations, without requiring any updates to the model parameters (Brown et al., 2020; Wei et al., 2022b; Liu et al., 2021; Lester et al., 2021). In practice, ICL typically uses a concatenation of a short sequence of annotated in-context examples known as *demonstrations* and a *test query* with its task-specific instruction. It offers a promising alternative to supervised fine-tuning since it greatly reduces the amount of required labeled data.

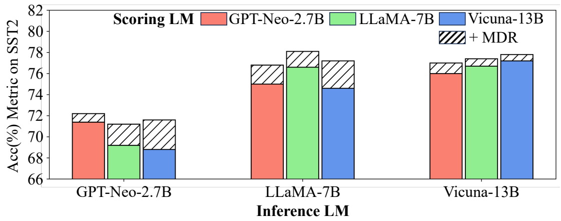

Figure 1: Experimental results of a tailored retriever trained on one LM and tested on an unseen dataset with different inference LLMs. The slashed box represents the improvement brought by MDR.

The quality of the selected demonstrations plays a crucial role in ICL (Liu et al., 2022; Min et al., 2022). To improve ICL's performance, given a test query, previous works mainly construct a demonstration pool and retrieve the most textual or semantically similar demonstrations by using BM25 (Robertson and Zaragoza, 2009), Sentence-BERT (Reimers and Gurevych, 2019) or other off-the-shelf sentence embeddings, which outperforms random selection. But these approaches are heuristic and sub-optimal as they lack guidance from task supervision. Alternatively, another approaches utilize the feedback signals of a Language Model (LM), referred to as the scoring LM, to identify positive and negative demonstrations. Subsequently, they employ a two-encoder retriever, where one encoder encodes the test query and the other encodes the demonstrations. The retriever is trained using contrastive learning to optimize the similarity between the test query and positive demonstrations,

*Equal Contribution.
†Corresponding Author.

Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 4189–4204
June 16-21, 2024 ©2024 Association for Computational Linguistics

simultaneously minimizing it for negative demonstrations. During inference, the demonstrations exhibiting the highest similarity are retrieved and concatenated with the test query to form the model input. This approach has shown better performance in specific task domains, such as question answering (Das et al., 2021) and semantic parsing (Rubin et al., 2022). Recent efforts have shifted towards cross-task prompt retrieval (Cheng et al., 2023) and unified multi-task retrieval (Li et al., 2023) by incorporating signals from various tasks across multiple data domains to train one retriever, which has demonstrated impressive improvements.

However, distinct LLMs exhibit different biases (Mao et al., 2023; Lu et al.) for "what is a good demonstration" since they possess differences in training data, model architectures and training methods. As a result, a demonstration suitable for one LLM may not be appropriate for others. We conclude that existing methods fail to retrieve the most appropriate demonstrations for different inference LLMs. There are two key factors: (i) **Bias Introduced in Training Phase:** Previous methods solely rely on one scoring LM to identify positive and negative demonstrations. The inherent bias of the scoring LM can impact this identification process, potentially introducing noise to the retriever training. Consequently, this may lead to the retrieval of bad demonstrations when inference on different LLMs. (ii) **Bias Neglected in Inference Phase:** A high-quality demonstration should not only exhibit similarity to the test query, but also align with the bias of the inference model. Previous methods employ the similarity-based extraction of demonstrations, which greatly neglect the inference model's bias. To test the bias issue, we train 3 tailored retrievers on RTE dataset using three different scoring LMs, including `GPT-Neo-2.7B`, `LLaMA-7B` and `Vicuna-13B`. Then we test the performance of each retriever on SST2 dataset under different inference LLMs, respectively. As shown in Fig 1, the one-to-one corresponding scoring LM and inference LLM perform the best while performances on different inference LLMs can be inferior, which illustrates that using a single scoring LM for various inference LLMs limits the performance of ICL. Since training multiple retrievers for multiple inference LLMs is costly, retrieving demonstrations that meet inference model's bias at inference stage is crucial to improve the ICL performance. However, directly estimating and evaluating the model bias is impractical, the evaluation of a demonstration's appropriateness for a specific inference LLM remains a challenging and under-explored aspect in ICL.

To address this limitation, we propose a simple yet effective eigenvalue-based evaluation metric to assess the appropriateness of demonstrations for a specific inference LLM. Based on this metric, we further propose MDR, a **M**odel-specific **D**emonstration **R**etrieval (MDR) method for ICL at inference time. Given a pre-trained demonstration retriever, MDR retrieves the most appropriate demonstrations from a pre-constructed demonstration pool for a given test query and a specific inference LLM. Results in Fig 1 verify that MDR is effective of mitigating the bias issue by reducing the performance gap between the corresponding and the non-corresponding scoring-inference LLMs.

To evaluate MDR's generalization across diverse models, we test MDR on a variety of seen and unseen tasks with different LLMs at inference time. The outcomes highlight the effectiveness of MDR as a simple and straightforward method to select the most appropriate demonstrations for a specific LLM without any additional training or human annotation.

Our contributions are summarized as follows:

- We introduce a novel evaluation metric to measure the appropriateness of demonstrations for a specific inference LLM.
- We propose MDR, a simple and effective framework to retrieve the most appropriate demonstrations for different inference LLMs, without any training cost.
- Experiments on 23 datasets across 11 data domains under various LLMs (2.7B $\sim$ 13B) show the validity and scalability of MDR.

2 Task Definition

We aim to improve the performance of ICL by retrieving the most appropriate demonstrations for any given task input. Specifically, given a pre-trained demonstration retriever, a test query x_{test} and an inference model LM, we retrieve the top-K most appropriate demonstrations $\{p_j\}_{j=1}^{K}$ from a pre-constructed demonstration pool $\mathbb{P}$. These demonstrations are concatenated with the test query in ICL fashion. Our objective is to optimize the model prediction y^*_{test} to align with the ground truth y_{test}, where y^*_{test} can be denoted by:

$$y^*_{test} = P_{\mathrm{LM}}(y^*_{test}|p_K \oplus ... \oplus p_1 \oplus x_{test}). \quad (1)$$

3 Preliminaries

We give a detailed introduction to the framework of previous retrieval-based methods in this section. These methods rely on a retriever to retrieve demonstrations from a demonstration pool. The retriever is based on a bi-encoder architecture (Rahman and Ng, 2012) containing two encoders E_q and E_p, where E_q encodes the test query and E_p encodes demonstrations. The training and inference phases of the retriever are outlined as follows.

3.1 Training Phase

Existing works usually use the feedback signal of a LM to distinguish positive and negative samples, and train the retriever with contrastive learning. The details are as follows.

To obtain the training data, a demonstration pool $\mathbb{P}$ is first constructed by selecting a certain number of data examples from multiple datasets covering various task types including Reading Comprehension, Closed-book QA, Paraphrase Detection, etc. Each data example, consisting of the input text and its task label, can be denoted as $e = (e_x, e_y)$.

For each data example $e \in \mathbb{P}$, a set of demonstration candidates $\{p_i\}_{i=1}^{L}$ is constructed by randomly selecting L data examples from $\mathbb{P}$. Then a specific model G, namely the scoring LM, is used to identify demonstration candidates as positive or negative. For each candidate $p_i \in \{p_i\}_{i=1}^{L}$, e_x is concatenated with p_i to form the input of G. According to the task type that e belongs to, the score of p_i respecting to e is calculated as follows:

$$\text{score}\left(p_i, e_x\right) = \text{metric}\left(e_y, e_y^*\right), \tag{2}$$

where $e_y^* = P_{\text{G}}\left(e_y^* | p_i \oplus e_x\right)$ is the prediction of the scoring LM and $\text{metric}()$ is the specific task metric, such as F1 or Rouge for generation tasks and accuracy for classification tasks. The demonstration candidate with the highest score is identified as the positive, denoted as p^+, while others are constructed into a negative examples set, denoted as $\{p_j^-\}_{j=1}^{L-1}$.

Based on the training data, previous works utilize InfoNCE loss (van den Oord et al., 2018) to maximize the similarity score between data example e and its positive demonstrations, while minimize it for negative demonstrations:

$$\begin{aligned} & L(e_x, p^+, p_1^-, \dots p_{L-1}^-) \\ = \; & -\log \frac{e^{\text{sim}(e_x, p^+)}}{e^{\text{sim}(e_x, p^+)} + \sum_{j=1}^{L-1} e^{\text{sim}(e_x, p_j^-)}}. \end{aligned} \tag{3}$$

The similarity is calculated by:

$$\text{sim}(e_x, p) = E_q(e_x)^\top E_p(p) \tag{4}$$

where E_q encodes e_x, E_p encodes the demonstration p, and $p \in \{p^+\} \cup \{p_j^-\}_{j=1}^{L-1}$.

3.2 Inference Phase

Given a test query x_{test}, the trained retriever is used to encode x_{test} and all demonstrations in $\mathbb{P}$, then generate their embeddings. Previous works use Maximum Inner-Product or FAISS (Johnson et al., 2021) to calculate the similarity between the embeddings of x_{test} and all demonstrations. As a result, top-K similar demonstrations are retrieved and then concatenated with the test query to form the input of inference models.

We consider that a high-quality demonstrations should not only exhibit similarity to the test query, but also align with the model bias. However, previous works greatly neglect this impact and fail to retrieve the most appropriate demonstrations for a specific inference LLM, resulting in a generalization degradation.

4 Methodology

Provided by a frozen pre-trained demonstration retriever $\mathcal{R}$ trained on a demonstration pool $\mathbb{P}$, and various inference models with their parameters, we aim to retrieve the most appropriate demonstrations from $\mathbb{P}$ for a test query to improve the ICL performance without any retriever parameters updating. To achieve this, in this section, we first introduce a novel evaluation metric to evaluate the appropriateness of a demonstration to a specific inference LLM. Subsequently, we propose an inference framework named MDR, to re-rank demonstrations with our proposed evaluation metric.

4.1 Evaluation Metric

Based on ICL, LLMs are able to perform various tasks conditioned on a few input-output demonstrations (Brown et al., 2020). Consequently, the quality of these demonstrations plays a crucial role in guiding LLMs to achieve the best generalization performance effectively (Liu et al., 2022). In this paper, we assume that a high-quality demonstration should not only exhibit similarity to the test query but also align with the preferences of the specific inference model, which is overlooked by previous methods. Given a demonstration $p = (x, y)$, one general indicator capable of reflecting the model's

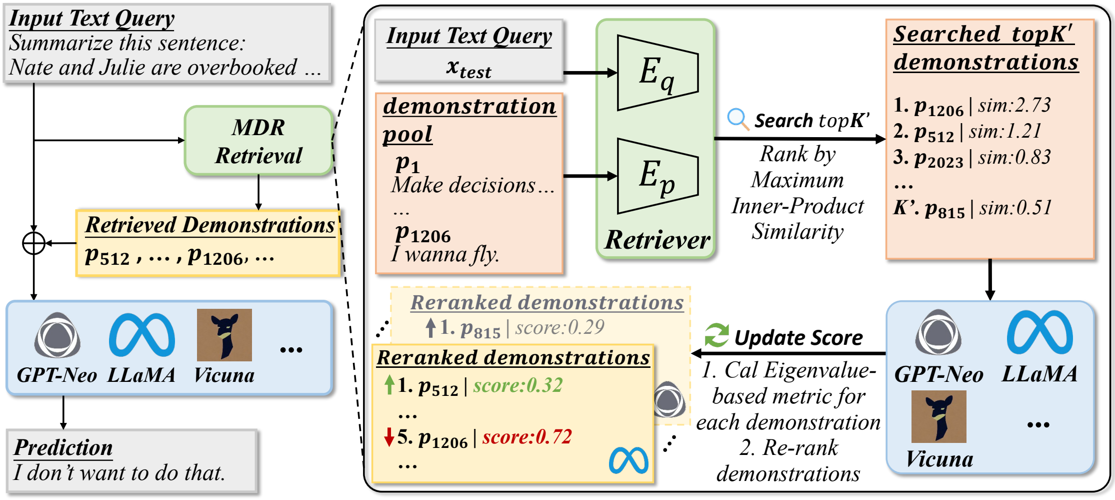

Figure 2: The inference framework of MDR. Given a test query, we first use the trained retriever to select top K' demonstrations from the demonstration pool based on Maximum Inner-Product. Then we score and re-rank these demonstrations with our proposed evaluation metric given a specific inference LLM. Finally, the top K demonstrations are concatenated with the test query to guide the inference model in making predictions.

preference for p is the loss $\mathcal{J}(y, y^*)$ computed between the ground truth y and the prediction y^* generated by the inference model when provided with x as input. We adopt the per-token negative log-likelihood as the loss function.

$$\mathcal{J}(y, y^*) = -\mathbb{E}_{y|x}[log\ p_\theta(y|x)] \tag{5}$$

$$y^* = P_{\text{LM}}(y^*|x). \tag{6}$$

While a smaller loss may suggest that the demonstration is better comprehended by the inference model, it is not conclusive evidence that it can serve as a better demonstration in ICL. As claimed in Zhuo et al. (2023) and Zheng and Saparov (2023), LLMs are vulnerable to perturbed prompts, such as minor typos, synonyms, or other common lexical perturbations. If a demonstration has minor typos, it is itself not robust to the model. Then, when it is fed to the model as part of the ICL prompt, it may cause LLMs to shift their focus towards the perturbed elements, thus producing wrong responses (Zhu et al., 2023). On the other hand, given that Language Generation Models generate words based on the previous context, an ICL demonstration carries a blend of contextual information across layers (Ferrando et al., 2023). If the demonstration is not robust to the model, its meaning space may be distorted by semantic perturbations. Such pollution potentially hampers the natural language understanding capabilities of the inference model (Zheng and Saparov, 2023), resulting in inferior model performance. Therefore, a model-preferred demonstration should not only exhibit a small loss but also demonstrate robustness to the perturbation. Due to the lack of labeled data describing the robustness of a demonstration to a specific inference model, the unsupervised evaluation of robustness can be extremely challenging.

To solve this, we are inspired by Zhao et al. (2019), who adopt the Fisher Information Matrix (FIM) of the input sample as a metric tensor to measure the robustness of deep learning models in adversarial attack task. Borrowing from this idea, we define a novel FIM-based matrix $\mathbf{H}$ to characterize the vulnerability of a demonstration to the perturbation in its feature space for a specific model LM. The matrix $\mathbf{H} \in \mathbb{R}^{n \times n}$ is defined as:

$$\mathbf{H} = \nabla_x \mathcal{J}(y, y^*)^\top \nabla_x \mathcal{J}(y, y^*), \tag{7}$$

where n is the hidden dimension of LM and $\nabla_x \mathcal{J}(y, y^*)$ is the partial differential of $\mathcal{J}(y, y^*)$ respecting to x.

Similar to the conclusion of Zhao et al. (2019), which quantifies the vulnerability of deep learning models, we deduce that the maximum eigenvalue of $\mathbf{H}$, denoted as λ_{max}, reflects the vulnerability of demonstrations to LM. A smaller λ_{max} indicates a more robust demonstration with higher resilience to the perturbation. The expression for λ_{max} can be written as:

$$\lambda_{max} = \frac{1}{m} \sum_{i=1}^{m} (\nabla_{x_i} \mathcal{J}(y, y^*))^2, \tag{8}$$

where m is the sequence length of x and x_i is the ith token.

We deduce four distinct scenarios based on the interplay between the loss and λ_{max}. A demonstration with a large loss and a large λ_{max} indicates a misfit with the model and should be discarded. Conversely, a demonstration with a large loss but a small λ_{max} suggests a potential misalignment between the model and the demonstration, possibly within a region of flatter gradient, or when the model is on the verge of a state where the gradient vanishes at this specific demonstration. On the other hand, a demonstration exhibiting a small loss but a large λ_{max} indicates a well-fitted model for this demonstration. But this data may be outliers or atypical samples. The model can exhibit high sensitivity to such demonstration, leading to significant output variants even with a slight change in the input space. Therefore, demonstrations with both a small loss and a small λ_{max} are considered as appropriate for LM. Subsequently, we propose a tuning coefficient C to integrate these two considerations. Given a demonstration $p = (x, y)$, we present the final expression of the evaluation metric, denoted as $\mathrm{Metric}(p)$:

$$\mathrm{Metric}(p) = C \cdot \lambda_{max} + (1 - C) \cdot \mathcal{J}(y, y^*). \quad (9)$$

In general, using the aforementioned derivation, a demonstration $p = (x, y)$ can be easily evaluated by three steps. Firstly, input x to the inference model and calculate the loss $J(y, y^*)$ using y and the model prediction y^*. Secondly, compute the maximum eigenvalue λ_{max} using Eq 8. Finally, $\mathrm{Metric}(p)$ is derived using $J(y, y^*)$ and λ_{max} according to Eq 9. demonstrations with a small $\mathrm{Metric}(p)$ are considered as appropriate for LM.

The significance of the introduced evaluation metric is that it remains invariant as long as the loss function keeps unchanged, ensuring stability and necessitating calculation only once. Moreover, the computation of $\mathrm{Metric}(p)$ is simple and convenient without any human or machine annotation.

4.2 Inference Framework

Based on the evaluation metric, we further propose a novel framework, MDR, to evaluate and retrieve the most appropriate demonstrations for a specific inference LLM. The overall inference framework is shown in Figure 2.

Given a test query x_{test} and a specific inference model LM, a high-quality demonstration should be appropriate to both x_{test} and LM. For the first consideration, in MDR retrieval stage, we use E_q to encode test query x_{test} and E_p to encode each demonstration in the demonstration pool $\mathbb{P}$, then retrieve K' demonstrations using Maximum Inner-Product similarity to construct a set of demonstration candidates $\{p_i\}_{i=1}^{K'}$:

$$\{p_i\}_{i=1}^{K'} = top\text{-}K'_{p_i \in \mathbb{P}} \, \mathrm{sim}(x_{test}, p_i). \quad (10)$$

For the second consideration, we calculate $\mathrm{Metric}(p_i)$ for demonstration $p_i = (x_i, y_i)$, $i = (1, 2, ..., K')$ using Eq 9. Then the K' demonstrations are re-ranked according to $\mathrm{Metric}(p)$ in ascending order and the top K ($K \leq K'$) demonstrations are selected:

$$\{p_j\}_{j=1}^{K} = top\text{-}K_{p_j \in \{p_i\}_{i=1}^{K'}} \, \mathrm{Metric}(p_j). \quad (11)$$

As a result, demonstrations with relatively lower similarity to the test query may have higher ranks due to their better appropriateness with LM. We then concatenate the top K demonstrations with the test query to form the model input $p_K \oplus ... \oplus p_1 \oplus x_{\text{test}}$.

5 Experiment Settings

5.1 Implementation Details

In our experiment, for the demonstration retriever, we adopt the pre-trained one from UPRISE (Cheng et al., 2023), which has been trained on a diverse set of tasks. The reason we choose UPRISE is that only UPRISE releases the checkpoint of the pre-trained retriever. Furthermore, CEIL (Ye et al., 2023), UDR (Li et al., 2023), and UPRISE are all built upon the framework introduced by EPR (Rubin et al., 2022). Given that UPRISE has been tested in a more comprehensive scenario, involving both seen and unseen datasets across 5 models, we have selected it as our primary baseline method. The training data of UPRISE contains 30 datasets, resulting in a demonstration pool of $224k$ demonstrations. At inference stage, we evaluate the performance of MDR under three inference LLMs with different scales, including `GPT-Neo-2.7B`, `LLaMA-7B`, and `Vicuna-13B`. We set the number K' of the demonstrations retrieved by similarity to be 20 and K of the concatenated demonstrations to be 3. For hyper-parameter C, we set it to be 0.9.

Classification Tasks

Task		***WSD***	***NLI***		***Multi Choice***		***Sentiment Classification***			**Avg.**
Dataset		WIC	WNLI	ANLI	WSC273	CMQA	Amazon	MR	RT	
Metric		acc↑	acc↑	acc↑	acc↑	acc↑	acc↑	acc↑	acc↑	acc↑
GPT-Neo (2.7B)	0-Shot	49.68	50.39	32.35	73.62	24.65	35.19	53.60	51.40	46.36
	Random	49.52	48.50	26.60	72.52	25.30	34.25	51.55	53.18	44.56
	BM25	50.94	50.86	31.80	68.86	24.45	34.05	52.55	65.10	47.33
	SBERT	50.00	51.02	31.90	67.76	25.05	35.44	52.84	67.72	47.72
	UPRISE	50.15	50.70	33.05	75.09	24.15	**36.75**	53.30	69.88	49.13
	MDR	**53.29**	**51.18**	**33.20**	**76.55**	**25.45**	**36.75**	**54.44**	**71.01**	**50.23**
LLaMA (7B)	0-Shot	47.96	50.86	33.25	76.92	24.95	30.15	61.05	52.25	47.17
	Random	50.40	49.60	33.40	76.92	25.20	30.20	59.59	56.66	47.75
	BM25	51.09	50.07	31.60	67.39	24.60	29.25	60.65	71.76	48.30
	SBERT	49.68	48.34	31.60	67.39	25.25	30.20	62.74	70.73	48.24
	UPRISE	47.49	48.97	33.15	79.85	24.90	32.75	71.50	75.70	51.79
	MDR	**52.66**	**51.02**	**33.70**	**81.31**	**25.60**	**34.75**	**71.75**	**77.20**	**53.50**
Vicuna (13B)	0-Shot	49.92	49.13	32.30	79.12	27.00	32.70	65.40	60.22	49.47
	Random	49.40	49.40	32.00	78.38	27.20	35.60	64.00	77.00	51.62
	BM25	49.21	48.97	33.00	71.79	28.59	33.20	70.39	86.60	52.72
	SBERT	48.90	43.30	33.00	70.69	25.20	33.00	72.20	84.60	51.36
	UPRISE	49.21	49.92	**33.20**	82.78	28.59	37.35	74.30	85.45	55.10
	MDR	**51.00**	**51.20**	**33.20**	**83.15**	**28.79**	**39.40**	**76.40**	**86.80**	**56.24**

Generation Tasks

Task		***Question Answering***				***Translation***		***Text Sum***			***Code Sum***	
Dataset		SQuADv2		Trivia QA		WMT14	WMT16	SamSum			Java	Python
Metric		em↑	f1(%)↑	em↑	f1(%)↑	bleu↑	bleu↑	r-1↑	r-2↑	r-l↑	bleu↑	bleu↑
GPT-Neo (2.7B)	0-Shot	0.10	0.56	0.20	1.34	2.15	2.44	11.60	0.02	11.57	6.91	6.86
	Random	6.50	9.27	0.75	3.15	19.63	23.40	11.33	0.11	11.32	6.98	6.63
	BM25	17.34	22.48	0.80	4.33	17.87	26.88	11.12	0.32	11.02	7.20	6.59
	SBERT	18.00	22.00	1.50	5.82	20.57	27.37	11.33	0.06	11.30	6.93	6.69
	UPRISE	31.25	39.60	1.60	5.97	18.79	24.68	11.50	1.09	11.37	7.42	6.91
	MDR	**32.55**	**40.30**	**2.40**	**8.79**	**22.55**	**29.04**	**11.94**	**1.31**	**11.73**	**7.51**	**6.97**
LLaMA (7B)	0-Shot	0.00	4.71	0.00	2.46	12.23	11.48	11.24	2.64	10.65	5.20	5.60
	Random	0.00	3.31	0.00	2.03	14.36	14.68	11.26	2.25	10.48	6.16	5.36
	BM25	**0.10**	6.04	0.00	2.08	14.28	15.45	12.78	3.63	11.76	7.27	5.79
	SBERT	0.00	5.90	0.00	2.40	16.03	15.87	11.38	2.54	10.50	5.88	5.66
	UPRISE	0.05	6.36	0.00	2.54	15.75	14.53	11.32	2.53	10.42	6.57	5.76
	MDR	0.00	**6.40**	0.00	**8.02**	**19.36**	**20.69**	**15.64**	**3.95**	**14.42**	**7.36**	**6.08**
Vicuna (13B)	0-Shot	0.00	3.06	0.20	10.40	8.99	28.69	11.59	2.36	10.90	21.63	15.84
	Random	**2.60**	9.70	**1.00**	5.00	23.50	24.03	19.71	4.51	17.89	23.87	8.42
	BM25	**2.60**	27.07	0.00	9.00	20.76	36.10	19.02	4.95	17.20	20.79	16.08
	SBERT	1.60	27.66	0.20	8.92	22.67	33.57	19.34	5.46	17.25	23.58	18.10
	UPRISE	0.60	27.59	0.00	10.37	29.65	35.83	20.30	**6.12**	18.46	25.66	19.83
	MDR	0.60	**27.95**	0.00	**10.62**	**36.07**	**37.89**	**20.83**	5.63	**18.87**	**26.57**	**20.22**

Table 1: Overall experimental results on unseen datasets. We report F1 score of Rouge-1 (r-1), Rouge-2 (r-2) and Rouge-L (r-l) on SamSum dataset.

5.2 Datasets

We choose 8 seen datasets from the demonstration pool and 15 unseen datasets across 8 task types to test MDR. All datasets are transformed into natural language instructions using randomly selected instruction templates from FLAN (Wei et al., 2022a) or UDR (Li et al., 2023). For each dataset, we report metrics on test set if available, falling back to the validation set otherwise. For detailed information, please refer to A.1.

5.3 Baselines

We compare MDR with previous methods. **0-Shot**: The input to the model is the concatenation of the task instruction and the test query only. **Random**: We random select the demonstrations. **BM25** (Robertson and Zaragoza, 2009): For each test query, we use BM25 to retrieve the most similar demonstrations. **SBERT** (Reimers and Gurevych, 2019): We use Sentence-Bert to retrieve the most similar demonstrations. **UPRISE** (Cheng et al., 2023): UPRISE is a recently proposed representative method for demonstration retrieval. For fair comparison, we test all baselines on each task under the same experimental settings.

6 Main Results

6.1 Overall

Performance on Unseen Datasets We show the performance comparison on unseen datasets where the demonstrations and the test query belong to different data domains. As shown in Table 1, MDR outperforms baselines significantly on most tasks across different inference LLMs. Specially, compared with 0-Shot, the accuracy of MDR is improved by up to 44.13% on classification task (MR) using `Vicuna-13B`. F1 score is increased by up to 39.74 points on generation task (SQuADv2) using `GPT-Neo-2.7B`. Moreover, when compared with UPRISE, the performance of MDR is improved by up to 10.8% and 41.2% for classification task (WIC) and generation task (SamSum) using `LLaMA-7B`, respectively. The results verify the effectiveness of MDR. However, in text generation tasks, especially in QA, the performance of the proposed solution does not achieve as high a level compared to the baselines as in other evaluation scenarios. We speculate that the task's inherent nature may contribute to this difference, and we intend to investigate this aspect further in future research.

Performance on Seen Datasets To make a comprehensive comparison of MDR on seen datasets where the demonstrations and the test query belong to the same data domain, we select 8 representative datasets covering multiple task types. As shown in Table 2, MDR exhibits a considerable advantage on most datasets. Specifically, MDR has an absolute improvement of up to 34.98% when compared with random selection on SQuADv1 dataset using `GPT-Neo-2.7B`, and a relative improvement of up to 6.3% when compared with UPRISE on COPA dataset using `LLaMA-7B`. The results prove the effectiveness of MDR.

6.2 Ablation Study

We perform an ablation study to assess the influence of various hyper-parameters, including: the number of top demonstrations retrieved by similarity, denoted by K'; the number of concatenated demonstrations, denoted by K; and the hyper-parameter C.

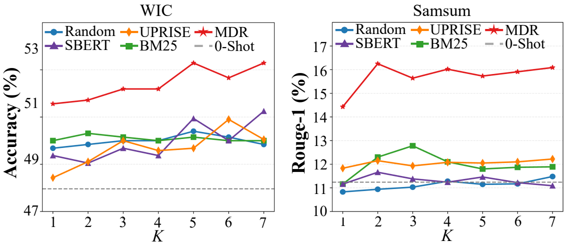

Figure 3: Ablation study on concatenated demonstrations number K. The experiments are conducted on WIC and SamSum with `LLaMA-7B`.

6.2.1 The impact of K.

To assess the impact of the concatenated demonstrations number, denoted by K, we conduct a comparative analysis involving random selection, BM25, SBERT, and UPRISE. This evaluation is performed on one classification task, i.e., WIC, and one generation task, i.e., SamSum, using `LLaMA-7B` while keeping other parameters unchanged. Specifically, K' is set to be 20. The results are shown in Figure 3. As K increases, the accuracy rises consistently, indicating that more examples can help the model do better ICL. Moreover, MDR demonstrates substantial advantages over the baseline methods, irrespective of the value of K.

6.2.2 The impact of K'.

The quantity of top demonstrations, initially selected based on similarity and subsequently evaluated through MDR, significantly influences model performance. To test the impact of K', we conduct experiments on four distinct datasets: WSC273 and RT for classification task, Trivia QA and SQuADv2 for generation task using `GPT-Neo-2.7B` while maintaining consistent parameter settings. Specifically, K is set to be 3. The results are shown in Figure 4. Notably, the highest accuracy is achieved at $K' = 20$. Consistent with the previous analysis, increasing the value of K' provides more opportunities for demonstrations that are less similar to the test query but are more compatible with the inference LLM. However, the performance of the model does not keep increasing as K increases. It is explainable since the demonstrations exhibiting low similarity to the test query may rank higher after re-ranking. Though these demonstrations may be more appropriate to the inference model, they are not appropriate to the test query. Therefore, it's essential to keep the balance between the similarity to the test query and the appropriateness to the model.

Task		*NLI*	*Comm Reasoning*		*Para Detection*		*Sentiment CLF*		*Reading Comprehension*		
Dataset		RTE	COPA	PIQA	MRPC		SST2	YELP	OBQA	SQuADv1	
Metric		acc↑	acc↑	acc↑	acc↑	f1↑	acc↑	acc↑	acc↑	em↑	f1↑
GPT-Neo (2.7B)	0-Shot	35.74	67.00	68.60	44.11	42.71	51.60	79.00	44.00	0.20	4.82
	Random	34.29	68.00	72.60	60.53	71.70	55.40	77.40	45.00	8.00	16.91
	UPRISE	62.81	71.00	72.80	73.28	80.28	78.60	93.20	49.20	38.20	49.51
	MDR	**66.06**	**75.00**	**75.40**	**76.47**	**83.61**	**81.00**	**93.80**	**49.80**	**41.80**	**51.89**
LLaMA (7B)	0-Shot	49.45	67.00	74.80	35.04	15.33	51.00	83.20	47.59	0.00	5.95
	Random	47.29	73.00	75.80	58.82	70.62	52.60	79.00	52.00	0.00	4.90
	UPRISE	62.81	79.00	77.60	72.54	79.41	83.60	96.60	56.80	**0.20**	7.21
	MDR	**65.34**	**84.00**	**78.40**	**76.43**	**83.67**	**84.20**	**97.39**	**58.80**	0.00	**7.62**
Vicuna (13B)	0-Shot	53.42	74.00	77.60	51.35	66.01	60.19	78.80	46.00	0.60	9.79
	Random	49.45	77.00	77.00	58.08	69.84	77.00	83.00	49.60	0.60	9.27
	UPRISE	53.06	86.00	79.00	69.36	77.63	89.20	94.80	61.60	0.60	9.47
	MDR	**55.23**	**90.00**	**81.80**	**70.58**	**78.57**	**90.40**	**95.60**	**62.40**	**0.60**	**10.70**

Table 2: Overall experimental results on seen datasets.

6.2.3 The joint impact of K and K'.

As K and K' jointly impact the final experimental results, we evaluate the model's performance on the WIC dataset using `GPT-Neo-2.7B` to assess how increasing both K and K' simultaneously on the same dataset affects the results. All other parameters remain unchanged. As observed in A.2 Table 7, increasing K from 4 to 10 only brings marginal improvement, aligning with findings in Li et al. (2023) and Chen et al. (2023). When $K \leq 10$, the optimal K' falls within the range of 15 to 20. Consequently, selecting $K' = 20$ proves to be the optimal choice for the retrieval-based ICL.

Additionally, we perform experiments by setting $K = K'$ on WSC273 dataset using `GPT-Neo-2.7B` (i.e., MDR only re-ranks the order of demonstrations). As shown in A.2 Table 4, MDR consistently outperforms baselines as K' increases, providing further validation of the effectiveness of MDR in evaluating and retrieving model-specific appropriate demonstrations for ICL.

6.2.4 The impact of C.

We apply parameter tuning method to explore the impact of C for different inference LLMs. Experiments are conducted with C from 0.0 to 1.0 on one classification dataset, i.e., WSC273, and one generation dataset, i.e., SQuADv2, while keeping other parameters unchanged. As shown in Fig 5, the best accuracy occurs when $C = 0.9$, indicating the importance of considering the appropriateness to the inference model when evaluating the quality of a demonstration. Additionally, models perform badly when $C = 0.0$ and $C = 1.0$, demonstrating that solely relying on the loss or the eigenvalue-based metric is inadequate to measure the appropri-

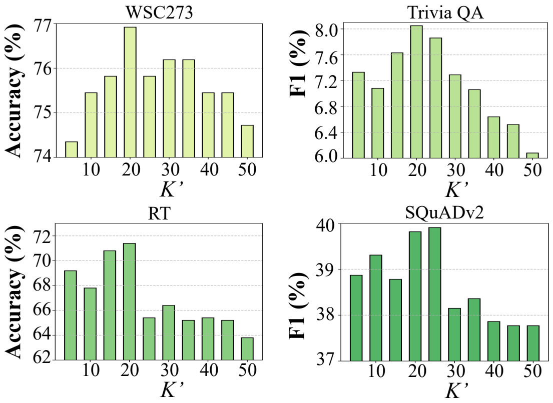

Figure 4: Ablation study on the number K' of demonstrations initially selected by similarity. The experiments are conducted on WSC273, Trivia QA, RT and SQuADv2 with `GPT-Neo-2.7B`.

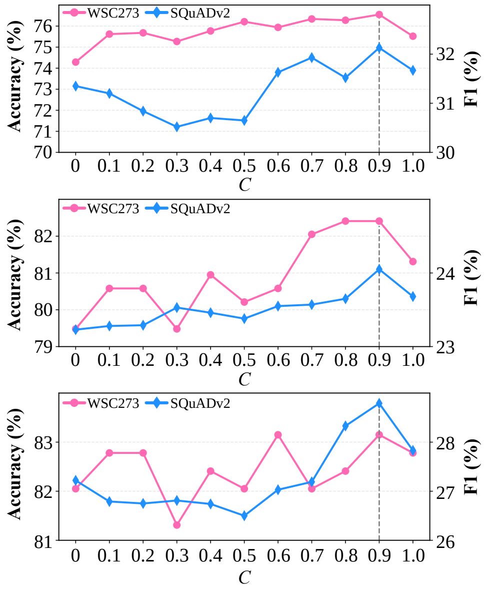

Figure 5: Ablation study on C. The experiments are conducted on WSC273 and SQuADv2 with `GPT-Neo-2.7B` (up), `LLaMA-7B` (middle) and `Vicuna-13B` (down).

ateness of a demonstration for a specific model.

We discover an erratic pattern when C is small, indicating that loss takes a large proportion in our metric. Given that loss itself is an uncertainty factor, the re-rank process can be influenced. This influence causes the quality of demonstrations to depend more on similarity in the first step. In such cases, the performance across different datasets can suffer from fluctuations since various datasets retrieve demonstrations with different extents of similarity. On the other hand, when C increases to a certain extent (such as between 0.7 and 1.0), the accuracy on both datasets across all three models consistently demonstrates a stable improvement. This illustrates that enhancing the proportion of robustness can make the re-rank process less sensitive to similarity and become more effective. In such a scenario, loss, robustness, and similarity can collaboratively work well. To further validate the pattern of C, we conduct experiments on two additional datasets using `GPT-Neo-2.7B` and `LLaMA-7B`. As shown in A.2 Table 8, the accuracy reaches a peak at 0.8 or 0.9, indicating that increasing C to a larger value reduces more unstable factors.

7 Related Work

In this section, we introduce previous works on exploring different strategies for selecting in-context demonstrations for LLMs. Two primary types of retrievers for ICL are commonly used. One is off-the-shelf retrievers, such as fine-tuned BERT (Liu et al., 2022), BM25 (Robertson and Zaragoza, 2009) or SBERT (Reimers and Gurevych, 2019). The other approaches train a task-specific retriever using designed signals, such as question answering (Das et al., 2021), code generation (Poesia et al., 2022) and dialogue state tracking (Hu et al., 2022). In particular, Rubin et al. (2022) introduce Efficient Prompt Retriever (EPR), which employs a LM to score demonstrations, and trains a dense retriever using contrastive learning. Additionally, Ye et al. (2023) propose CEIL to select diverse and helpful ICL demonstrations by using determinantal point processes. Recently, Li et al. (2023) propose UDR, a unified retriever designed for a wide range of tasks. Unlike UDR, which focuses on testing on seen tasks, Cheng et al. (2023) utilize a single retriever for cross-task and cross-model scenario on unseen tasks. However, previous studies neglect the biases of different inference LLMs. In this paper, we fully consider LLMs biases and strive to retrieve the most appropriate demonstrations for a specific inference model during inference stage.

8 Conclusion

In this paper, we introduce MDR, a simple yet effective method to evaluate and retrieve the model-specific appropriate demonstrations for ICL without re-training the retriever. MDR offers the feasibility of evaluating the appropriateness of a demonstration for LLMs without supervision. Moreover, given a trained retriever, MDR makes it more applicable to use it on various larger inference models.

In summary, MDR offers a promising insight for enhancing LLMs performance by leveraging their biases, presenting a notable perspective over existing retrieval-based methods and even prompt engineering.

Ethical Considerations

We believe that this study contributes intellectual value to the dependable application of retrieval-based in-context learning in the field of NLP, with potential broader implications for tasks in other areas. It is noteworthy that there are no direct societal consequences, and all experiments are conducted on open datasets in this work.

Limitations

Given the constraints of computing power, incorporating language models with larger scales poses a challenge for us. Despite selecting numerous and diverse tasks in this paper, the chosen set remains limited. Moreover, while results using automatic metrics provide a fair assessment of task performance, we aim to conduct a human evaluation in the near future.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under, Grant 62201072, Grant U23B2001, Grant 62171057, Grant 62101064, Grant 62001054, and Grant 62071067; in part by the Ministry of Education and China Mobile Joint Fund under Grant MCM20200202 and Grant MCM20180101; in part by the BUPT-China Mobile Research Institute Joint Innovation Center.

References

Luisa Bentivogli, Bernardo Magnini, Ido Dagan, Hoa Trang Dang, and Danilo Giampiccolo. 2009. The fifth PASCAL recognizing textual entailment challenge. In *TAC*. NIST.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: reasoning about physical commonsense in natural language. In *AAAI*, pages 7432–7439. AAAI Press.

Ond rej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Ale s Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*.

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

Jiuhai Chen, Lichang Chen, Chen Zhu, and Tianyi Zhou. 2023. How many demonstrations do you need for in-context learning? In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 11149–11159. Association for Computational Linguistics.

Daixuan Cheng, Shaohan Huang, Junyu Bi, Yuefeng Zhan, Jianfeng Liu, Yujing Wang, Hao Sun, Furu Wei, Weiwei Deng, and Qi Zhang. 2023. UPRISE: universal prompt retrieval for improving zero-shot evaluation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 12318–12337. Association for Computational Linguistics.

Rajarshi Das, Manzil Zaheer, Dung Thai, Ameya Godbole, Ethan Perez, Jay Yoon Lee, Lizhen Tan, Lazaros Polymenakos, and Andrew McCallum. 2021. Case-based reasoning for natural language queries over knowledge bases. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 9594–9611. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *IWP@IJCNLP*. Asian Federation of Natural Language Processing.

Javier Ferrando, Gerard I. Gállego, Ioannis Tsiamas, and Marta R. Costa-jussà. 2023. Explaining how transformers use context to build predictions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 5486–5513. Association for Computational Linguistics.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*.

Yushi Hu, Chia-Hsuan Lee, Tianbao Xie, Tao Yu, Noah A. Smith, and Mari Ostendorf. 2022. In-context learning for few-shot dialogue state tracking. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2627–2643. Association for Computational Linguistics.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos QA: Machine reading comprehension with contextual commonsense reasoning. 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)*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with gpus. *IEEE Trans. Big Data*, 7(3):535–547.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. triviaqa: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. *arXiv e-prints*, page arXiv:1705.03551.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *EMNLP (1)*, pages 3045–3059. Association for Computational Linguistics.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *KR*. AAAI Press.

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 *ACL 2023*, pages 4644–4668. Association for Computational Linguistics.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for gpt-3? In *DeeLIO@ACL*, pages 100–114. Association for Computational Linguistics.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021. GPT understands, too. *CoRR*, abs/2103.10385.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp.

Rui Mao, Qian Liu, Kai He, Wei Li, and Erik Cambria. 2023. The biases of pre-trained language models: An empirical study on prompt-based sentiment analysis and emotion detection. *IEEE Trans. Affect. Comput.*, 14(3):1743–1753.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *EMNLP*, pages 2381–2391. Association for Computational Linguistics.

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 *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 11048–11064. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial nli: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*.

Mohammad Taher Pilehvar and José Camacho-Collados. 2019. Wic: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1267–1273. Association for Computational Linguistics.

Gabriel Poesia, Alex Polozov, Vu Le, Ashish Tiwari, Gustavo Soares, Christopher Meek, and Sumit Gulwani. 2022. Synchromesh: Reliable code generation from pre-trained language models. In *ICLR 2022*. OpenReview.net.

Altaf Rahman and Vincent Ng. 2012. Resolving complex cases of definite pronouns: The winograd schema challenge. In *EMNLP-CoNLL*, pages 777–789. ACL.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *EMNLP*, pages 2383–2392. The Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *arXiv e-prints*, page arXiv:1606.05250.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *EMNLP/IJCNLP (1)*, pages 3980–3990. Association for Computational Linguistics.

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*. AAAI.

Ohad Rubin, Jonathan Herzig, and Jonathan Berant. 2022. Learning to retrieve prompts for in-context learning. In *NAACL-HLT*, pages 2655–2671. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, pages 1631–1642. ACL.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022a. Finetuned language models are zero-shot learners. In *ICLR*. OpenReview.net.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, Quoc Le, and Denny Zhou. 2022b. Chain of thought prompting elicits reasoning in large language models. *CoRR*, abs/2201.11903.

Jiacheng Ye, Zhiyong Wu, Jiangtao Feng, Tao Yu, and Lingpeng Kong. 2023. Compositional exemplars for in-context learning. *CoRR*, abs/2302.05698.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*, pages 649–657.

Chenxiao Zhao, P. Thomas Fletcher, Mixue Yu, Yaxin Peng, Guixu Zhang, and Chaomin Shen. 2019. The adversarial attack and detection under the fisher information metric. In *AAAI2019, IAAI 2019, EAAI 2019*, pages 5869–5876. AAAI Press.

Hongyi Zheng and Abulhair Saparov. 2023. Noisy exemplars make large language models more robust: A domain-agnostic behavioral analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 4560–4568. Association for Computational Linguistics.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Neil Zhenqiang Gong, Yue Zhang, and Xing Xie. 2023. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. *CoRR*, abs/2306.04528.

Terry Yue Zhuo, Zhuang Li, Yujin Huang, Fatemeh Shiri, Weiqing Wang, Gholamreza Haffari, and Yuan-Fang Li. 2023. On robustness of prompt-based semantic parsing with large pre-trained language model: An empirical study on codex. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 1090–1102. Association for Computational Linguistics.

A Appendix

A.1 Datasets

We give detailed split statistics and evaluation metric for each test datasets. The datasets are collected following FLAN (Wei et al., 2022a) and UDR (Li et al., 2023). we select 15 unseen datasets across 8 task types and various data domains from FLAN (Wei et al., 2022a) and UDR (Li et al., 2023) to evaluate the performance of MDR , including:

- **Word Sense Disambiguation:** WIC (Pilehvar and Camacho-Collados, 2019).
- **Natural Language Inference:** WNLI (Wang et al., 2019) and ANLI (Nie et al., 2020).
- **Multi Choice:** WSC273 (Levesque et al., 2012) and Cosmos QA (Huang et al., 2019).
- **Sentiment Classification:** Amazon (Li et al., 2023), MR (Li et al., 2023) and Rotten tomatoes (Pang and Lee, 2005).
- **Question Answering:** SQuADv2 (Rajpurkar et al., 2016) and Trivia QA (Joshi et al., 2017).
- **Translation:** WMT14 (Bojar et al., 2014) and WMT16 (Bojar et al., 2016).
- **Text Summarization:** Samsum (Gliwa et al., 2019).
- **Code Summarization:** Java (Li et al., 2023) and Python (Li et al., 2023).

For test efficiency, we limit the test set size of each dataset. Specifically, we randomly sample a 2000 subset for whose test set size is $\geq$ 2000 when testing on `GPT-Neo-2.7B` and 500 for whose test set size is $\geq$ 500 when testing on `Vicuna-13B`. We randomly sample a 2000 subset for classification tasks and a 500 subset for generation tasks when testing on `LLaMA-7B`. The detailed split statistics and evaluation metric are shown in Table 3.

Additionally, we select 8 representative datasets from the demonstration pool as the seen datasets, including:

- **Natural Language Inference:** RTE (Bentivogli et al., 2009).
- **Commonsense Reasoning:** COPA (Roemmele et al., 2011) and PIQA (Bisk et al., 2020).
- **Paraphrase Detection:** MRPC (Dolan and Brockett, 2005).
- **Sentiment Classification:** SST2 (Socher et al., 2013) and YELP (Zhang et al., 2015).
- **Reading Comprehension:** OBQA (Mihaylov et al., 2018) and SQuADv1 (Rajpurkar et al., 2016).

Similarly, we limit the test set size of each dataset to be 500. The detailed split statistics and evaluation metric are shown in Table 5.

We set the maximum sequence generation length to be 100. The test process of `GPT-Neo-2.7B` can be implemented with only one 3090-24GB. The memory occupied by `LLaMA-7B` and `Vicuna-13B` is below 70GB, which means that all tests can be completed with a single A100-80GB.

A.2 Ablation Study

We present the experimental results in this section.

A.3 Case Study

To further demonstrate the effectiveness of MDR, we present the rankings of demonstrations retrieved using the Maximum Inner-Product method and the rankings after re-ordering by

Cluster	Task	Report Split	TestSize (GPT-Neo-2.7B)	TestSize (LLaMA-7B)	TestSize (Vicuna-13B)	Metric
WSD	WIC	validation	638	638	500	acc
NLI	WNLI	train	635	635	500	acc
	ANLI	train_r1	2000	2000	500	acc
Multi Choice	WSC273	test	273	273	273	acc
	CMQA	validation	2000	2000	500	acc
Sentiment Classification	Amazon	test	2000	2000	500	acc
	MR	test	2000	2000	500	acc
	RT	test	1070	1070	500	acc
Question Answering	SQuADv2	test	2000	500	500	em
	Trivia QA	validation	2000	500	500	em f1
Translation	WMT14	test	2000	500	500	bleu
	WMT16	test	2000	500	500	bleu
Text Sum	SamSum	test	819	500	500	rouge-1 rouge-2 rouge-l
Code Sum	Java	test	2000	500	500	bleu
	Python	test	2000	500	500	bleu

Table 3: The statistics, split and evaluation metrics of unseen datasets.

K=K'	UPRISE	MDR
0	72.52	
3	**75.82**	**75.82 (+0.00)**
5	76.19	**76.55 (+0.36)**
7	76.92	**77.65 (+0.73)**
9	76.55	**78.30 (+1.75)**
11	76.92	**77.65 (+0.73)**

Table 4: Experimental results on WSC273 dataset using GPT-Neo-2.7B when $K = K'$.

MDR (Table 6). The results demonstrate a significant influence of MDR on rankings. Firstly, demonstration No.144251, initially ranked second using similarity-based retrieval, shifts to the sixth position after re-ranking when applied to GPT-Neo-2.7B, and further drops beyond the seventh rank on LLaMA-7B. Secondly, the rankings of the top demonstrations exhibit notable variations. These results emphasize that demonstrations with high similarity to the test query do not necessarily align with the model bias. On the contrary, demonstrations with lower similarity still show the potential to enhance their rankings after re-ranking by MDR. Consequently, it becomes crucial to consider model biases during demonstrations retrieval. MDR furnishes a direct insight into this aspect of the field.

More specifically, we present the results for a task query tested on GPT-Neo-2.7B and LLaMA-7B using UPRISE and MDR, respectively. As shown in Table 9, for a given test query, UPRISE retrieves the same demonstrations for different inference LLMs, resulting in varying predictions. In contrast, when using MDR, as shown in Table 10, MDR retrieves different demonstrations for different inference models, all of which yield correct answers. This confirms the effectiveness of MDR.

Cluster	Task	Report Split	TestSize (GPT-Neo-2.7B)	TestSize (LLaMA-7B)	Metric
NLI	RTE	validation	500	500	acc
Commonsense Reasoning	COPA	validation	500	500	acc
	PIQA	validation	500	500	acc
Paraphrase Detection	MRPC	validation	500	500	acc f1
Sentiment Classification	SST2	validation	500	500	acc
	YELP	test	500	500	acc
Reading Comprehension	OBQA	test	500	500	acc

Table 5: The statistics, split and evaluation metrics of seen datasets.

Infer Model	Retrieval Phase	Prompt Attribute	Rank#1 Prompt	Rank#2 Prompt	Rank#3 Prompt	Rank#4 Prompt	Rank#5 Prompt	Rank#6 Prompt	Rank#7 Prompt
/	Similarity-based Search	id task	160353 yelp	144251 sst2	160717 yelp	159436 yelp	147132 sst2	147879 sst2	148630 sst2
GPT-Neo (2.7B)	Eigenvalue-based Re-rank	id task score	145331 sst2 3.2532	147894 sst2 3.2605	146698 sst2 3.2899	145985 sst2 3.2945	147132 sst2 3.3019	144251 sst2 3.3052	148200 sst2 3.3152
LLaMA (7B)	Eigenvalue-based Re-rank	id task score	147844 sst2 0.3886	145985 sst2 0.3929	146698 sst2 0.398	151484 sst2 0.3996	147894 sst2 0.4005	149971 sst2 0.4042	147132 sst2 0.4069

Table 6: The first line is the top-7 demonstrations extracted by Maximum Inner-Product. The second and third lines represent the top-7 demonstrations after re-ranking with our proposed eigenvalue-based metric under GPT-Neo-2.7B and LLaMA-7B.

K\K’	5	10	15	20	25	30	35	40	45	50
0	49.68									
1	52.19	51.80	**52.50**	52.19	52.35	51.72	51.09	50.94	51.09	51.25
2	52.29	52.97	53.44	**54.07**	52.97	53.91	53.91	53.60	53.60	50.94
3	52.97	53.29	53.76	**54.38**	52.97	53.76	53.44	53.44	53.29	53.13
4	53.76	53.29	**54.07**	53.29	53.29	52.19	52.35	53.29	52.66	52.82
5	52.51	53.61	53.13	**54.86**	53.61	53.13	53.61	53.13	53.13	52.82
6	/	53.29	**53.76**	53.29	52.35	53.44	52.66	53.44	53.44	53.60
7	/	54.23	53.76	**54.38**	52.19	53.13	52.97	53.29	53.13	53.13
10	/	53.44	53.13	**54.07**	52.35	52.19	52.66	52.66	53.13	53.13
12	/	/	**54.07**	53.13	52.82	53.13	52.19	52.35	52.97	53.44
14	/	/	**54.07**	53.13	53.29	53.13	51.88	52.19	53.13	53.13
16	/	/	/	52.66	52.82	**53.13**	52.35	52.03	52.82	52.50
18	/	/	/	52.82	**53.65**	52.97	53.13	52.35	52.50	52.97
20	/	/	/	52.66	**53.44**	53.29	52.82	52.03	52.82	53.29

Table 7: The joint ablation study of parameters K and K' on WIC dataset using GPT-Neo-2.7B.

Model	C	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
GPT-Neo (2.7B)	**WIC**	51.00	50.52	50.30	50.78	51.07	51.30	51.40	51.97	52.23	**53.29**	53.12
	RT	70.39	70.36	70.31	70.22	70.31	70.44	70.57	70.61	**71.10**	71.01	70.97
LLaMA (7B)	**WCI**	51.72	51.75 |	51.94	51.78	51.82	51.88	51.94	52.02	52.31	**52.66**	52.62
	RT	75.32	76.00	75.80	75.40	75.80	76.50	76.40	76.60	77.10	**77.20**	76.90

Table 8: Additional ablation study of C on WIC and RT datasets with GPT-Neo-2.7B and LLaMA-7B.

UPRISE

Dataset: Amazon
Retrieved example numbers: 3

Test Query
Sentiment of the sentence: The first time I used this product it was fabulous. My hair turned out beautifully. When I tried another day to get the same results, the curling iron wouldn’t heat up. It is

Ground Truth
terrible

Retrieved demonstrations
Q: What is the sentiment of the following review? “Their Chocolate is divine; so creamy and smooooth! Whether you get a Pure Chocolate or a Blended Mocha you will be in Chocolate Heaven. I have even bought the chocolate powder for home and it never disappoints. It is so nice to shop the district with a mocha or pure chocolate in hand. I cannot walk by without stopping in :-)"
A: Positive

Q: “can only love the players it brings to the fore for the gifted but no-nonsense human beings they are and for the still-inestimable contribution they have made to our shared history." How would the sentiment of this sentence be perceived?
A: Positive

Q: Is the following review positive or negative? “Sara is so talented. She new exactly what I wanted and she gave me the best hair cut ever. I will use her from now on."
A: Positive

Prediction of GPT-Neo-2.7B:
terrible

Prediction of LLaMA-7B:
great

Table 9: The results of the case study for a task query tested on `GPT-Neo-2.7B` and `LLaMA-7B` when using UPRISE.

MDR

Dataset: Amazon
Retrieved example numbers: 3
Test Query
Sentiment of the sentence: The first time I used this product it was fabulous. My hair turned out beautifully. When I tried another day to get the same results, the curling iron wouldn’t heat up. It is
Ground Truth
terrible

GPT-Neo-2.7B

Retrieved demonstrations
Q: “both heartbreaking and heartwarming ... just a simple fable done in an artless sytle , but it ’s tremendously moving." How would the sentiment of this sentence be perceived?
A: Positive

Q: “filling nearly every minute ... with a lighthearted glow , some impudent snickers , and a glorious dose of humankind ’s liberating ability." How would the sentiment of this sentence be perceived?
A: Positive

Q: “a journey spanning nearly three decades of bittersweet camaraderie and history , in which we feel that we truly know what makes holly and marina tick , and our hearts go out to them as both continue to negotiate their imperfect , love-hate relationship." How would the sentiment of this sentence be perceived?
A: Positive

Prediction of GPT-Neo-2.7B:
terrible

LLaMA-7B

Retrieved demonstrations
Q: “both heartbreaking and heartwarming ... just a simple fable done in an artless sytle , but it ’s tremendously moving." How would the sentiment of this sentence be perceived?
A: Positive

Q: “both a beautifully made nature film and a tribute to a woman whose passion for this region and its inhabitants still shines in her quiet blue eyes." How would the sentiment of this sentence be perceived?
A: Positive

Q: “muccino , who directed from his own screenplay , is a canny crowd pleaser , and the last kiss ... provides more than enough sentimental catharsis for a satisfying evening at the multiplex . Ḧow would the sentiment of this sentence be perceived?
A: Positive

Prediction of LLaMA-7B:
terrible

Table 10: The results of the case study for a task query tested on GPT-Neo-2.7B and LLaMA-7B when using MDR.