Beyond Semantic Similarity: Multi-Stage Calibration for In-Context Learning

Chunling Xi[1], Di Liang[2*]
[1]Pacific Insurance Technology Co., Ltd. [2]Lixin Information Services (Shenzhen) Co., Ltd.
xichunling987@163.com

***Abstract*—In-Context Learning (ICL) empowers Large Language Models (LLMs) to adapt to downstream tasks by conditioning on a few demonstrations. However, the efficacy of ICL is heavily contingent on the quality of selected examples. Existing methods predominantly rely on semantic similarity for retrieval, which suffers from two critical misalignments: (1) Distributional Mismatch: colloquial user queries often diverge from the canonical representation of demonstrations, impairing retrieval accuracy; and (2) Utility Gap: semantically similar examples do not necessarily contribute positively to the model's reasoning, and may even introduce noise. To address these challenges, we propose Multi-Stage Calibration (MSC), a framework designed to align the retrieval process with the generation objective. MSC operates in three stages: first, it refines noisy queries into canonical forms via a perplexity-guided rewrite mechanism; second, it introduces a contribution-aware re-ranker that explicitly learns to predict the marginal utility of each demonstration rather than mere similarity; finally, it employs a generator-guided aggregation strategy to identify the optimal context. Extensive experiments across multiple benchmarks demonstrate that MSC significantly outperforms standard retrieval baselines, offering a robust solution for deploying ICL in open-domain scenarios.**

***Index Terms*—In-context learning, Large language model, Neural language processing**

I. Introduction

Large Language Models (LLMs) have sparked a paradigm shift in natural language processing, demonstrating emergent capabilities across a spectrum of complex reasoning tasks [1]. A cornerstone of this success is In-Context Learning (ICL), a mechanism that enables models to adapt to novel downstream tasks by conditioning on a natural language instruction and a handful of task-specific demonstrations, without the need for computationally expensive parameter updates. This "learning to learn" ability has made ICL the de facto standard for deploying LLMs in few-shot scenarios.

However, the efficacy of ICL is far from robust. Recent studies indicate that LLM performance is notoriously volatile and heavily contingent on the selection of in-context examples [5]. A suboptimal set of demonstrations can lead to catastrophic performance drops, hallucinated reasoning, or format violations. Consequently, the challenge of identifying the most effective demonstrations from a large candidate pool has become a focal point of research. Standard approaches predominantly rely on unsupervised dense retrieval (e.g., Sentence-BERT), operating on the assumption that semantic similarity serves as a reliable proxy for demonstration utility. While intuitive, we argue that this similarity-based retrieval paradigm suffers from two fundamental limitations that create a Retrieval-Generation Gap: **1) The Query-Distribution Mismatch:** In real-world deployments, user queries are frequently colloquial, ambiguous, or noise-laden. In contrast, high-quality demonstration pools typically consist of canonical, well-structured text. This distributional shift confuses dense retrievers, which operate on embedding spaces sensitive to surface-level lexical features. As a result, the retriever may fetch examples that share keywords with the noisy query but lack structural relevance to the underlying task logic. **2) The Similarity-Utility Paradox:** A central but often flawed assumption in existing work is that $\text{Similarity}(x, x') \propto \text{Utility}(x' \rightarrow x)$. However, empirical evidence suggests this relationship is non-linear and occasionally inverse. An example might be semantically close to the query but contain a misleading reasoning path or a different answer distribution, acting as a "distractor" that induces negative transfer. Conversely, a less similar example might possess a crucial reasoning pattern that unlocks the model's potential. Pure similarity metrics fail to capture this intrinsic *task utility*.

To bridge these gaps, we propose **Multi-Stage Calibration (MSC)**, a unified framework designed to align the retrieval process with the generation objective. MSC emulates a human-like cognitive strategy: clarifying the question, evaluating the usefulness of references, and synthesizing the best information. Specifically, our framework operates in three progressive stages. First, Query Calibration projects colloquial inputs into the model's preferred distribution space via a perplexity-minimization rewrite mechanism, mitigating the distribution mismatch. Second, we introduce a Contribution-Aware Re-ranker, a lightweight module explicitly trained to predict the marginal utility of a demonstration (i.e., how much it increases the probability of the correct answer) rather than mere semantic overlap. Finally, recognizing that even top-ranked candidates may contain redundancy, Aggregation Calibration leverages the LLM's self-reflective capabilities to filter the context and identify the single most informative example.

Our contributions are summarized as follows: Firstly, we identify and formalize the critical limitations of similarity-based retrieval in ICL: the *distributional mismatch* and the *utility gap*. Secondly, we propose MSC, a novel coarse-to-fine framework that systematically refines the input query, the demonstration ranking, and the final context selection. Then, we introduce a utility-based scoring mechanism that effectively learns to distinguish beneficial demonstrations from distractors, shifting

the focus from semantic matching to functional contribution. Finally, extensive experiments demonstrate that MSC consistently outperforms strong retrieval baselines, offering a robust solution for open-domain ICL.

II. RELATED WORK

Prompting and LLM Reasoning The success of large language models has sparked interest in using prompting techniques to improve task performance [1]. While recent research has focused on task-specific instruction tuning, either by fine-tuning the entire model [24] or maintaining task-specific parameters [10], our work is a general prompting approach that can improve contextual learning ability without any fine-tuning. The use of language models to generate intermediate steps has been widely validated in the context of solving reasoning tasks, whether through training [9], zero-shot [2], or few-shot prompting [11]. Recent work has focused on problem decomposition and teaching language models to answer subtasks, ultimately answering complex questions [3]. MSC is orthogonal to these methods, enhancing input queries rather than assigning generation. Therefore, it can be easily extended by any of these prompting strategies. A closely related research direction involves exploring the interpretability and consistency of the fundamental principles generated by large models. Recent works [12] have improved the interpretability of numerical tasks, while [4] have improved the performance of various arithmetic and logical reasoning tasks by leveraging the consistency between multiple generated principles.

Demonstration Selection in-context learning (ICL) primarily retrieves relevant input/output pairs for a provided test example from an annotated dataset. [5] proposes to utilize dense semantic embedders to retrieve relevant examples to improve ICL. [13] Example of retrieving machine translation ICL using BM25. [6] explore knowledge-based question answering and dialogue state tracking respectively, and design specialized target similarity to train demonstration retrieval. [14] propose to train demonstration retrieval via feedback from LM and demonstrate its effectiveness on semantic parsing. [13] further explores cross-language semantic parsing using similar training signals as [14]. [15] uses random labels to retrieve relevant examples and proposes heuristics to reduce the negative impact of incorrect labels. Recently, [16] proposed UDR, a unified demonstration retriever for various NLP tasks, which is trained on approximately 40 annotated datasets with unified LM feedback. Although these methods rely on high-quality annotated datasets and only explore contextual learning without underlying principles, But MSC allows LLM to improve itself without annotated datasets and parameter updates.

Model Augmentation In addition, some works attempt to use data generated by LLMs for model enhancement. [7], [25]–[27] propose ZeroGen, ProGen, and ZeroGen$^+$ to use LLM-generated datasets to enhance small models such as LSTM or DistilBERT. [17], [28], [29], [34] suggest using LLMs to generate reasoning paths and teach small language models to reason. [18], [30]–[33] and [19], [35]–[40] use LLMs to generate instruction data and improve the LLM's instruction tracking ability. [20], [41]–[44] proposes ToolFormer, which learns how to use various tools through self-generated data. [8] and [9], [45]–[52] use LLMs to generate reasoning paths and improve themselves using annotated and unannotated datasets, respectively. [21] have LLMs generate COT data themselves. [22] uses LLMs to generate knowledge bases and improve their open-domain QA capabilities. These works reveal the importance of promoting high-quality reasoning in the CoT process. However, generating and utilizing high-quality examples that fit the LLM distribution makes this process challenging. This prompted us to design a better mechanism (MSC) to prompt language models in in-context learning. It is a new prompting scheme that allows LLM to output high-quality answers in a structured way through query calibration, recall calibration, and aggregation calibration.

III. METHODOLOGY

A. System Architecture

The Multi-Stage Calibration framework is designed to optimize the selection and presentation of in-context demonstrations by aligning them with the intrinsic preferences of Large Language Models (LLMs). The framework systematically addresses the retrieval-generation gap that often hinders In-Context Learning in real-world scenarios. As shown in the overall pipeline, MSC processes the input and demonstration candidates through three progressive calibration stages. First, the framework refines the input query to resolve distributional mismatches. Second, it evaluates potential demonstrations based on their marginal functional utility rather than surface-level similarity. Finally, it employs a generator-guided selection mechanism to finalize the context. This hierarchical approach ensures that the resulting prompt is both distributionally aligned and functionally optimal for the target LLM.

B. Query Distribution Refinement

User-issued queries in practical applications frequently exhibit high perplexity due to informal syntax, typos, or redundant noise. Such artifacts lead to a significant distributional mismatch between the raw input q_{raw} and the structured, well-formatted demonstrations stored in the vector database $\mathcal{D}$. To mitigate this shift, we first treat the LLM as a probabilistic re-writer. Given a task-agnostic instruction $\mathcal{P}_{calib}$, we sample a set of m candidate canonical queries:

$$\mathcal{Q}_{cand} = \{\hat{q}^{(1)}, \ldots, \hat{q}^{(m)}\} \sim P_{\mathcal{M}}(\cdot|\mathcal{P}_{calib}, q_{raw}) \quad (1)$$

where $P_{\mathcal{M}}$ represents the likelihood under the target generator. This process projects the colloquial input onto a linguistic manifold that is more familiar to the model's pre-training distribution. To ensure the most effective version is selected, we utilize Perplexity (PPL) as a proxy for distributional alignment. We evaluate each $\hat{q}^{(i)}$ and select the candidate that minimizes the average negative log-likelihood:

$$q_{calib} = \underset{\hat{q}\in\mathcal{Q}_{cand}}{\operatorname{argmin}} \left[\frac{1}{|\hat{q}|} \sum_{t=1}^{|\hat{q}|} -\log P_{\mathcal{M}}(w_t|w_{<t}) \right] \quad (2)$$

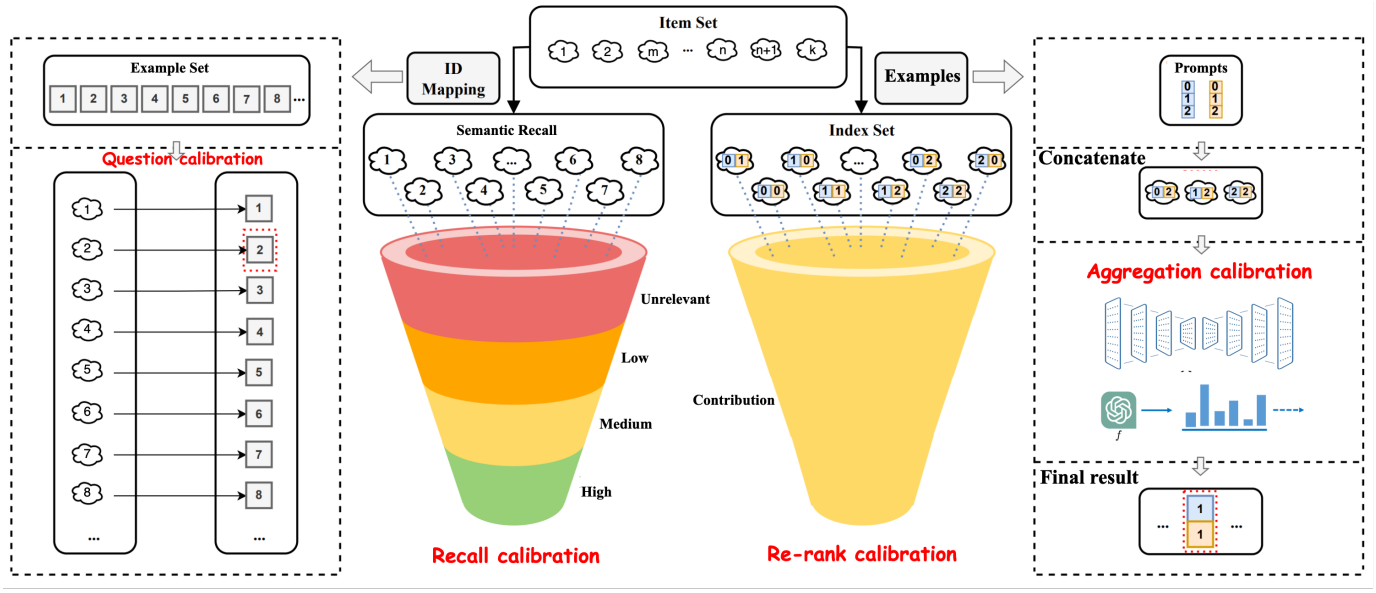

Fig. 1. The MSC framework consists of three modules: 1) Question calibration is used to calibrate the user input questions; 2) Recall calibration and Re-rank calibration is used to select high-quality example which most relevant to the question are recalled from the semantic level; 3) Aggregation calibration is used to Filter the most relevant examples from an LLM perspective.

This operation ensures that subsequent retrieval and inference are conditioned on a text representation that resides in the high-probability region of the model's embedding space, thereby enhancing latent alignment.

C. Utility-Aware Demonstration Selection

Traditional retrieval-based ICL relies on the assumption that semantic similarity is a sufficient proxy for demonstration effectiveness. However, semantically similar examples can occasionally act as distractors if they contain conflicting reasoning paths. We propose a two-step selection process centered on Pointwise Marginal Utility. First, a dense retriever is used for coarse filtering to reduce the search space to a candidate set $\mathcal{E}_{recall}$ containing the Top-N examples based on cosine similarity:

$$\mathcal{E}_{recall} = \text{Top-N}_{e \in \mathcal{D}}[\cos(\phi(q_{calib}), \phi(e))] \tag{3}$$

To identify which candidates truly facilitate the model's reasoning, we define the Ground-Truth Utility $u(e, q)$ of an example e as the log-likelihood gain of the correct answer y^*:

$$u(e, q) = \log P_{\mathcal{M}}(y^*|q_{calib}, e) - \log P_{\mathcal{M}}(y^*|q_{calib}, \emptyset) \tag{4}$$

We then train a lightweight cross-encoder re-ranker f_θ to approximate this score. The re-ranker is optimized using a regression objective over a training set $\mathcal{D}_{train}$ constructed by sampling query-example pairs and computing their utility via Eq. 4. The model is trained to minimize the Mean Squared Error (MSE):

$$\mathcal{L}(\theta) = \sum_{(q,e,u) \in \mathcal{D}_{train}} ||f_\theta(q \oplus e) - u||^2 \tag{5}$$

During inference, we re-rank $\mathcal{E}_{recall}$ and select the top-K examples $\mathcal{E}_{rank}$ with the highest predicted contribution for final context construction.

D. Generator-Guided Aggregation

The final stage of MSC addresses the potential for mutual interference or semantic redundancy when multiple high-utility demonstrations are combined. We utilize the self-reflective capabilities of the LLM to perform a final denoising of the re-ranked candidates. Given the query q_{calib} and the candidates $\mathcal{E}_{rank}$, we provide a selection prompt $\mathcal{P}_{judge}$ that instructs the LLM to act as an internal judge. The model identifies the most optimal demonstration index k^*:

$$k^* = \underset{k \in [1,K]}{\text{argmax}} P_{\mathcal{M}}(\text{"Select } k\text{"} | \mathcal{P}_{judge}, \mathcal{E}_{rank}, q_{calib}) \tag{6}$$

By leveraging the model's own feedback loop, we effectively eliminate candidates that may have bypassed the re-ranker due to surface-level lexical correlations but do not align with the model's specific reasoning logic. The final answer $\hat{y}$ is then generated using the single most effective demonstration:

$$\hat{y} = \text{Generate}(e_{k^*} \oplus q_{calib}) \tag{7}$$

This integrated calibration process ensures that the ICL prompt is both distributionally aligned and functionally beneficial, leading to enhanced performance in complex downstream tasks.

IV. Evaluation Setup

A. Benchmarks

We evaluate MSC on multiple benchmarks across four NLP task types, including perturbed queries. Numerical Reasoning GSM8K, SVAMP, AQUA-RAT, SingleOp and MultiArith, MMLU high school subset, and GSMIC-4k. Logical Reasoning Shuffled Objects, Date Understanding, LogiQA, and Coin Flipping. Reading Comprehension DROP subsets (Football, Non-football, Census, Break)and SQuAD. Commonsense Reasoning StrategyQA and Winogrande. We use Code-davinci-002 as

TABLE I
MSC EXPERIMENTAL RESULTS ON DIFFERENT LLMS AND DATASETS, INCLUDING ZERO-SHOT AND COT.

Model	Dataset	Zero-shot				Few-shot			
		Standard		CoT		Standard		CoT	
	MSC	False	True	False	True	False	True	False	True
Code-davinci-002	GSM8K	16.4	22.6(+5.2)	49.3	54.1(+4.8)	19.2	23.5(+4.3)	61.1	68.6(+7.5)
	SVAMP	66.8	74.1(+7.3)	66.5	74.3(+7.8)	69.8	76.5(+6.7)	75.2	81.3(+6.1)
	MultiArith	31.0	48.8(+17.8)	76.0	81.6(+5.6)	44.0	58.8(+14.8)	96.1	98.6(+2.5)
	SingleOp	91.6	92.3(+0.7)	82.9	92.7(+9.8)	93.2	94.6(+1.4)	92.8	95.1(+2.3)
	Shuffled Objects	36.4	37.4(+1.0)	42.4	58.9(+15.5)	34.8	36.6(+1.8)	66.0	69.7(+3.7)
	Coin Flip	47.7	47.6(−0.1)	58.5	62.1(+3.6)	99.6	100(+0.4)	100	100(+0.0)
	Date	44.2	43.9(−0.3)	39.0	47.4(+8.4)	49.3	51.0(+1.7)	65.6	69.8(+4.2)
	DROP (Football)	50.8	59.5(+8.7)	44.1	60.4(+16.3)	63.7	70.6(+6.9)	67.3	73.8(+6.5)
	DROP (Nonfootball)	43.2	57.3(+14.1)	39.7	52.7(+13.0)	57.1	64.5(+7.4)	69.2	74.2(+5.0)
	DROP (Census)	45.9	64.3(+18.4)	30.0	52.3(+22.3)	56.8	66.9(+10.1)	69.6	77.4(+7.8)
	DROP (Break)	43.7	56.9(+13.1)	38.2	52.5(+14.3)	55.5	63.7(+8.2)	65.3	69.7(+4.4)
	SQuAD (F1)	65.7	68.9(+3.2)	52.6	54.8(+2.2)	88.7	91.9(+3.2)	90.5	90.9(+0.4)
	AQUA-RAT	21.2	23.4(+2.2)	37.0	36.4(−0.6)	30.3	31.9(+1.6)	43.7	44.3(+0.6)
	MMLU-h	31.8	36.6(+4.8)	42.5	42.7(+0.2)	36.7	39.3(+2.6)	44.1	43.7(−0.4)
	logiQA	42.5	42.1(−0.4)	37.0	40.7(+3.7)	45.3	46.9(+1.6)	40.9	41.3(+0.4)
GPT-3.5 (Turbo)	GSM8K	5.6	24.8(+19.2)	75.7	77.6(+1.9)	31.3	34.5(+3.2)	75.1	84.4(+9.3)
	SVAMP	51.9	76.4(+24.5)	80.5	83.8(+3.3)	76.1	78.4(+2.3)	77.4	82.3(+5.9)
	MultiArith	76.5	83.7(+7.2)	93.4	96.6(+3.2)	83.4	90.5(+7.1)	97.8	98.8(+1.0)
	SingleOp	92.6	96.8(+4.2)	91.4	94.8(+3.4)	93.9	96.8(+2.0)	95.7	96.8(+1.1)
	Shuffled Objects	26.9	26.7(−0.2)	79.5	82.5(+3.0)	30.6	36.4(+5.8)	68.8	75.2(+6.4)
	Coin Flip	76.7	86.8(+10.1)	99.8	99.8(+0.0)	90.0	95.6(+5.6)	100	100(+0.0)
	Date	45.7	45.1(+0.4)	46.6	47.0(+0.4)	50.4	51.3(+1.9)	64.5	66.8(+2.3)
	DROP (Break)	47.1	52.8(+5.7)	51.9	51.7(−0.2)	59.9	62.7(+2.8)	61.6	66.7(+5.1)
	SQuAD (F1)	79.1	80.6(+1.5)	62.1	59.4(−2.7)	76.4	84.0(+7.6)	85.3	86.8(+1.5)
	AQUA-RAT	27.9	28.6(+0.7)	51.1	51.0(−0.1)	33.4	35.8(+2.4)	39.7	57.6(+17.9)
	MMLU-h	25.6	31.5(+5.9)	51.1	53.3(+2.2)	34.1	36.4(+2.2)	28.9	41.1(+12.2)
	logiQA	36.2	38.5(+2.3)	37.6	39.0(+1.4)	45.1	44.6(−0.5)	32.5	32.3(−0.2)

the primary model, with results on GPT-3.5-Turbo for subset validation.

V. MODEL PERFORMANCE

As we focus on determining whether the MSC approach can effectively enhance the reasoning precision or effectiveness for the LLMs, we extensively compared our method with zero-shot, zero-shot CoT, few-shot, and few-shot CoT prompt strategies. **Table I** provides the overall results of MSC. Firstly, based on the results of Code-davinci-002, We find that MSC significantly outperforms baselines on most datasets, which shows MSC's best comprehensive reasoning capability on a series of NLP tasks. It is noteworthy that the MSC exhibits a significant enhancement in zero-shot scenarios, especially for tasks with long query contexts, such as the DROP and SQuAD datasets. We observed an 18.4% improvement in accuracy for the zero-shot prompt DROP (census subset) dataset. Similarly, MSC using Zero-shot-CoT achieved an accuracy improvement (7.3%) on SVAMP, which makes the overall accuracy comparable to the few-shot CoT prompt. At the same time, MSC also achieved improvements when the baseline methods could not solve the task. For example, in the Shuffled Objects task involving three objects, MSC showed a slight improvement in zero-shot performance (from 36.4% to 37.4%). Nevertheless, it greatly improved the accuracy of Zero-shot-CoT (from 42.4% to 58.9%). Secondly, we also evaluated the effect of MSC using GPT-3.5-Turbo on a subset of experimental tasks. Overall, the results are consistent with our previous experiments on code-davinci-002. For example, MSC significantly improves the accuracy on GSM8K, from 75.1% to 84.4% .

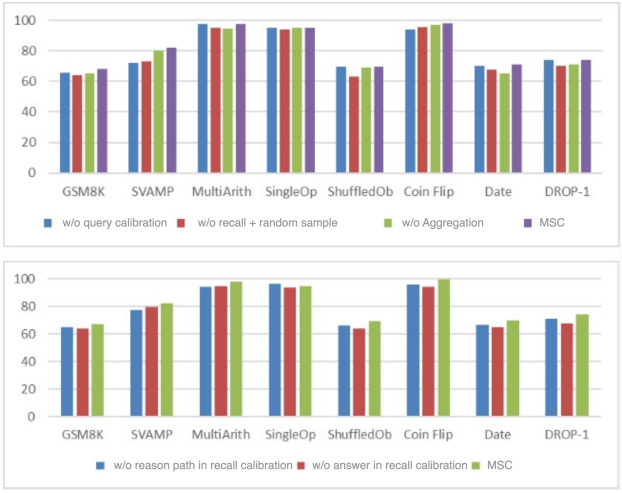

Fig. 2. Results of component ablation experiment.

A. Ablation Study

To evaluate the contribution of each individual component within The MSC, we conduct a series of ablation experiments across diverse reasoning benchmarks. As illustrated in Figure 2, several key observations emerge. First, the exclusion of the Query Distribution Refinement module leads to a substantial performance degradation across all tasks. This decline confirms

Method	GSM8K	DROP	MMLU
Decoding Paths=8			
Zero-Shot-CoT$_{T=0.7}$	63.2	56.9	31.8
Zero-Shot-CoT$_{T=1}$	63.9	57.4	32.4
Zero-Shot-CoT$_{T=1.2}$	63.5	53.8	32.2
Few-Shot-CoT$_{T=0.7}$	73.7	69.5	39.7
Few-Shot-CoT$_{T=1}$	73.6	70.3	39.8
Few-Shot-CoT$_{T=1.2}$	73.4	70.4	40.2
MSC$_{T=0.7}$	**86.5**	76.0	**43.6**
MSC$_{T=1}$	84.9	**76.5**	42.5
MSC$_{T=1.2}$	86.3	76.3	42.4
Decoding Paths=16			
Zero-Shot-CoT$_{T=0.7}$	68.6	60.7	33.8
Zero-Shot-CoT$_{T=1}$	69.2	59.6	35.3
Zero-Shot-CoT$_{T=1.2}$	67.9	58.5	34.7
Few-Shot-CoT$_{T=0.7}$	72.3	71.1	41.4
Few-Shot-CoT$_{T=1}$	73.5	72.4	41.3
Few-Shot-CoT$_{T=1.2}$	71.4	71.9	40.9
MSC$_{T=0.7}$	86.3	**76.7**	43.7
MSC$_{T=1}$	**86.4**	76.4	**44.2**
MSC$_{T=1.2}$	86.1	76.6	44.1

TABLE II
PERFORMANCE COMPARISON ON SELF-CONSISTENCY ACROSS DIFFERENT DECODING TEMPERATURES AND PATHS.

the existence of a significant distributional gap between raw user inputs and the model's optimal canonical space, highlighting that query calibration is essential for accurate intent interpretation. Second, while Zero-Shot-CoT occasionally outperforms Few-Shot-CoT on specific datasets suggesting that arbitrary or irrelevant demonstrations can introduce semantic noise MSC consistently maintains a performance lead over both. This demonstrates that MSC is not merely adding context, but is performing high-utility retrieval. Furthermore, the comparison with random sampling baselines shows that MSC's utility-aware selection aligns with human intuition, where semantically and functionally related demonstrations provide the most effective reasoning blueprints. Finally, removing the Aggregation Calibration module results in a measurable drop in accuracy. This indicates that even after high-quality re-ranking, the generator-guided selection acts as a crucial final filter to resolve potential redundancies and identify the most synergistic demonstration for the specific test instance.

VI. ANALYSIS

We evaluate the robustness of MSC under the Self-Consistency strategy [18] by varying the sampling temperature T and the number of decoding paths. The results in Table II demonstrate that MSC consistently outperforms both Zero-Shot-CoT and Few-Shot-CoT across all temperature settings and computational budgets. This stability indicates that the benefits of MSC are orthogonal to the gains provided by majority voting. We observe that the marginal improvement of MSC slightly narrows as the number of decoding paths increases from 8 to 16. This phenomenon can be attributed to the fact that a higher decoding budget encourages broader stochastic exploration of the solution space. In such cases, the LLM may explore unconventional reasoning paths where the influence of a fixed in-context demonstration becomes less dominant. However, MSC still retains the highest absolute accuracy, proving its reliability in budget-constrained scenarios.

VII. CONCLUSION

In this paper, we introduced Multi-Stage Calibration (MSC), a unified framework designed to alleviate the retrieval-generation gap in In-Context Learning. By systematically calibrating user queries, estimating demonstration utility, and performing generator-guided aggregation, MSC effectively aligns informal user inputs with the model's reasoning requirements. Extensive experiments across multiple benchmarks spanning numerical reasoning, logic, and reading comprehension demonstrate that MSC significantly enhances LLM performance in an unsupervised manner. Crucially, MSC achieves these gains without the need for task-specific annotations or parameter updates. Our findings suggest that the functional utility of a demonstration is as important as its semantic similarity, and that multi-stage calibration is a robust path toward more reliable and adaptive few-shot learners in open-domain environments.

REFERENCES

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Zieglxer, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," 2020.

[2] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," *arXiv preprint arXiv:2205.11916*.

[3] D. Zhou, N. Schärli, L. Hou, J. Wei, N. Scales, X. Wang, D. Schuurmans, C. Cui, O. Bousquet, Q. Le, and E. Chi, "Least-to-most prompting enables complex reasoning in large language models," 2022.

[4] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," *arXiv preprint arXiv:2305.10601*, 2023.

[5] J. Liu, D. Shen, Y. Zhang, B. Dolan, L. Carin, and W. Chen, "What makes good in-context examples for gpt-3?" *arXiv preprint arXiv:2101.06804*, 2021.

[6] R. Das, M. Zaheer, D. Thai, A. Godbole, E. Perez, J.-Y. Lee, L. Tan, L. Polymenakos, and A. McCallum, "Case-based reasoning for natural language queries over knowledge bases," *arXiv preprint arXiv:2104.08762*.

[7] J. Ye, J. Gao, Q. Li, H. Xu, J. Feng, Z. Wu, T. Yu, and L. Kong, "Zerogen: Efficient zero-shot learning via dataset generation," *arXiv preprint arXiv:2202.07922*, 2022.

[8] J. Huang, S. S. Gu, L. Hou, Y. Wu, X. Wang, H. Yu, and J. Han, "Large language models can self-improve," *arXiv preprint arXiv:2210.11610*.

[9] E. Zelikman, Y. Wu, J. Mu, and N. Goodman, "Star: Bootstrapping reasoning with reasoning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 15 476–15 488, 2022.

[10] X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th IJCNN (Volume 1: Long Papers)*, Aug. 2021, pp. 4582–4597. [Online]. Available: https://aclanthology.org/2021.acl-long.353

[11] J. Wei, X. Wang, D. Schuurmans, M. Bosma, E. Chi, Q. Le, and D. Zhou, "Chain of thought prompting elicits reasoning in large language models," *arXiv preprint arXiv:2201.11903*, 2022.

[12] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," 2023.
A. Madaan and A. Yazdanbakhsh, "Text and patterns: For effective chain of thought, it takes two to tango," *arXiv preprint arXiv:2209.07686*, 2022.

[13] S. Agrawal, C. Zhou, M. Lewis, L. Zettlemoyer, and M. Ghazvininejad, "In-context examples selection for machine translation," *arXiv preprint arXiv:2212.02437*, 2022.

[14] O. Rubin, J. Herzig, and J. Berant, "Learning to retrieve prompts for in-context learning," *arXiv preprint arXiv:2112.08633*, 2021.

[15] X. Lyu, S. Min, I. Beltagy, L. Zettlemoyer, and H. Hajishirzi, "Z-icl: Zero-shot in-context learning with pseudo-demonstrations," *arXiv preprint arXiv:2212.09865*, 2022.

[16] X. Li, K. Lv, H. Yan, T. Lin, W. Zhu, Y. Ni, G. Xie, X. Wang, and X. Qiu, "Unified demonstration retriever for in-context learning," *arXiv preprint arXiv:2305.04320*, 2023.

[17] N. Ho, L. Schmid, and S.-Y. Yun, "Large language models are reasoning teachers," *arXiv preprint arXiv:2212.10071*, 2022.

[18] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," *arXiv preprint arXiv:2203.11171*, 2022.

[19] O. Honovich, T. Scialom, O. Levy, and T. Schick, "Unnatural instructions: Tuning language models with (almost) no human labor," *arXiv preprint arXiv:2212.09689*, 2022.

[20] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language models can teach themselves to use tools," *arXiv preprint arXiv:2302.04761*, 2023.

[21] Z. Shao, Y. Gong, Y. Shen, M. Huang, N. Duan, and W. Chen, "Synthetic prompting: Generating chain-of-thought demonstrations for large language models," *arXiv preprint arXiv:2302.00618*, 2023.

[22] J. Li, Z. Zhang, and H. Zhao, "Self-prompting large language models for open-domain qa," *arXiv preprint arXiv:2212.08635*, 2022.

[23] B. Wang and A. Komatsuzaki, "GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model," https://github.com/kingoflolz/mesh-transformer-jax, May 2021.

[24] Chang Dai, Hongyu Shan, Mingyang Song, and Di Liang. Hope: Hyperbolic rotary positional encoding for stable long-range dependency modeling in large language models. *arXiv preprint arXiv:2509.05218*, 2025.

[25] Zichu Fei, Qi Zhang, Tao Gui, Di Liang, Sirui Wang, Wei Wu, and Xuan-Jing Huang. Cqg: A simple and effective controlled generation framework for multi-hop question generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6896–6906, 2022.

[26] Ziyuan Gao, Di Liang, Xianjie Wu, Philippe Morel, and Minlong Peng. Decorl: Decoupling reasoning chains via parallel sub-step generation and cascaded reinforcement for interpretable and scalable rlhf. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 30789–30797, 2026.

[27] Tao Gui, Qi Zhang, Jingjing Gong, Minlong Peng, Di Liang, Keyu Ding, and Xuan-Jing Huang. Transferring from formal newswire domain with hypernet for twitter pos tagging. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2540–2549, 2018.

[28] Bo Li, Di Liang, and Zixin Zhang. Comateformer: Combined attention transformer for semantic sentence matching. *arXiv preprint arXiv:2412.07220*, 2024.

[29] Junchen Li, Chao Qi, Rongzheng Wang, Qizhi Chen, Liang Xu, Di Liang, Bob Simons, and Shuang Liang. When safety becomes a vulnerability: Exploiting llm alignment homogeneity for transferable blocking in rag. *arXiv preprint arXiv:2603.03919*, 2026.

[30] Liang Li, Qisheng Liao, Meiting Lai, Di Liang, and Shangsong Liang. Local and global: Text matching via syntax graph calibration. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11571–11575. IEEE, 2024.

[31] Di Liang, Fubao Zhang, Qi Zhang, and Xuan-Jing Huang. Asynchronous deep interaction network for natural language inference. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2692–2700, 2019.

[32] Di Liang, Fubao Zhang, Weidong Zhang, Qi Zhang, Jinlan Fu, Minlong Peng, Tao Gui, and Xuanjing Huang. Adaptive multi-attention network incorporating answer information for duplicate question detection. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, pages 95–104, 2019.

[33] Peiyang Liu, Ziqiang Cui, Di Liang, and Wei Ye. Who stole your data? a method for detecting unauthorized rag theft. *arXiv preprint arXiv:2510.07728*, 2025.

[34] Xiaoyu Liu, Xiaoyu Guan, Di Liang, and Xianjie Wu. Dpi: Exploiting parameter heterogeneity for interference-free fine-tuning. *arXiv preprint arXiv:2601.17777*, 2026.

[35] Xiaoyu Liu, Di Liang, Hongyu Shan, Peiyang Liu, Yonghao Liu, Muling Wu, Yuntao Li, Xianjie Wu, Li Miao, Jiangrong Shen, et al. Structural reward model: Enhancing interpretability, efficiency, and scalability in reward modeling. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 672–685, 2025.

[36] Yonghao Liu, Mengyu Li, Di Liang, Ximing Li, Fausto Giunchiglia, Lan Huang, Xiaoyue Feng, and Renchu Guan. Resolving word vagueness with scenario-guided adapter for natural language inference. *arXiv preprint arXiv:2405.12434*, 2024.

[37] Yonghao Liu, Di Liang, Fang Fang, Sirui Wang, Wei Wu, and Rui Jiang. Time-aware multiway adaptive fusion network for temporal knowledge graph question answering. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[38] Yonghao Liu, Di Liang, Mengyu Li, Fausto Giunchiglia, Ximing Li, Sirui Wang, Wei Wu, Lan Huang, Xiaoyue Feng, and Renchu Guan. Local and global: Temporal question answering via information fusion. In *IJCAI*, pages 5141–5149, 2023.

[39] Ruotian Ma, Yiding Tan, Xin Zhou, Xuanting Chen, Di Liang, Sirui Wang, Wei Wu, Tao Gui, and Qi Zhang. Searching for optimal subword tokenization in cross-domain ner. *arXiv preprint arXiv:2206.03352*, 2022.

[40] Qi Qian, Muling Wu, Zisu Huang, Wenhao Liu, Changze Lv, Xiaohua Wang, Zhenghua Wang, Zhengkang Guo, Zhibo Xu, Lina Chen, et al. Adaptive curriculum strategies: Stabilizing reinforcement learning for large language models.

[41] Jian Song, Di Liang, Rumei Li, Yuntao Li, Sirui Wang, Minlong Peng, Wei Wu, and Yongxin Yu. Improving semantic matching through dependency-enhanced pre-trained model with adaptive fusion. *arXiv preprint arXiv:2210.08471*, 2022.

[42] Rongzheng Wang, Yihong Huang, Muquan Li, Jiakai Li, Di Liang, Bob Simons, Pei Ke, Shuang Liang, and Ke Qin. Rethinking llm-driven heuristic design: Generating efficient and specialized solvers via dynamics-aware optimization. *arXiv preprint arXiv:2601.20868*, 2026.

[43] Sirui Wang, Di Liang, Jian Song, Yuntao Li, and Wei Wu. Dabert: Dual attention enhanced bert for semantic matching. *arXiv preprint arXiv:2210.03454*, 2022.

[44] Yao Wang, Di Liang, and Minlong Peng. Not all parameters are created equal: Smart isolation boosts fine-tuning performance. *arXiv preprint arXiv:2508.21741*, 2025.

[45] Muling Wu, Qi Qian, Wenhao Liu, Xiaohua Wang, Zisu Huang, Di Liang, LI Miao, Shihan Dou, Changze Lv, Zhenghua Wang, et al. Progressive mastery: Customized curriculum learning with guided prompting for mathematical reasoning. *arXiv preprint arXiv:2506.04065*, 2025.

[46] Xianjie Wu, Di Liang, Jian Yang, Xianfu Cheng, LinZheng Chai, Tongliang Li, Liqun Yang, and Zhoujun Li. Breaking size barrier: Enhancing reasoning for large-size table question answering. In *International Conference on Database Systems for Advanced Applications*, pages 241–256. Springer, 2025.

[47] Xianjie Wu, Xiaohang Xu, Tingyu Jiang, Jian Yang, Di Liang, Xianfu Cheng, Zhenhe Wu, Linzheng Chai, Wei Zhang, Jiaheng Liu, et al. Mmtablebench: A multi-level multimodal benchmark for reasoning and layout complexity in table qa. In *Proceedings of the ACM Web Conference 2026*, pages 3881–3892, 2026.

[48] Xianjie Wu, Jian Yang, Linzheng Chai, Ge Zhang, Jiaheng Liu, Xeron Du, Di Liang, Daixin Shu, Xianfu Cheng, Tianzhen Sun, et al. Tablebench: A comprehensive and complex benchmark for table question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25497–25506, 2025.

[49] Xianjie Wu, Jian Yang, Tongliang Li, Shiwei Zhang, Yiyang Du, LinZheng Chai, Di Liang, and Zhoujun Li. Unleashing potential of evidence in knowledge-intensive dialogue generation. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[50] Chao Xue, Di Liang, Pengfei Wang, and Jing Zhang. Question calibration and multi-hop modeling for temporal question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19332–19340, 2024.

[51] Chao Xue, Di Liang, Sirui Wang, Jing Zhang, and Wei Wu. Dual path modeling for semantic matching by perceiving subtle conflicts. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[52] Rui Zheng, Rong Bao, Yuhao Zhou, Di Liang, Sirui Wang, Wei Wu, Tao Gui, Qi Zhang, and Xuanjing Huang. Robust lottery tickets for pre-trained language models. *arXiv preprint arXiv:2211.03013*, 2022.