Efficient Demonstration Selection by Label-Alignment Divergence Reranking for In-Context Learning

Anonymous ACL submission

Abstract

In-context learning (ICL) performance is highly sensitive to which demonstrations are selected. Most existing selectors rely on semantic similarity, which can retrieve label-conflicting examples under ambiguity or noisy demonstration pools, leading to degraded performance. We propose **LADR** (**L**abel-**A**ligned **D**ivergence **R**eranking), a two-stage framework that augments TopK retrieval with label-distribution alignment. LADR fine-tunes a BERT-like classifier to estimate label distributions for the test input and retrieved candidates, and reranks them using Jensen-Shannon divergence. Candidate-side distributions are computed and cached offline, making inference-time reranking lightweight. Across seven benchmarks and multiple LLM families and scales, LADR consistently outperforms strong baselines. LADR is also robust to label permutation and reversal, as well as out-of-domain demonstration pools, and achieves a favorable accuracy-efficiency trade-off. The code is released here: `https://anonymous.4open.science/r/L2D-401B`

1 Introduction

In-context learning (ICL) (Brown et al., 2020) is an emergent capability of large language models (LLMs), enabling them to make predictions from a small set of input-output demonstrations provided at inference time (Dong et al., 2024). Compared to zero-shot prompting, ICL often yields stronger performance and has been widely applied to a broad range of NLP tasks(Jiang and Wang, 2025; Xu et al., 2024). Despite these advances, ICL remains highly sensitive to the choice (and order) of demonstrations (Min et al., 2022; Liu et al., 2022), motivating a large body of work on automatic demonstration selection.

A dominant paradigm selects demonstrations by semantic similarity between the test input and candidates (Liu et al., 2022; Rubin et al., 2022; Peng et al., 2024; Zhang et al., 2025b). While semantic relevance is important, it implicitly assumes that semantic proximity implies label consistency. In practice, this assumption is often violated: two inputs can be semantically similar yet carry conflicting labels due to negation, contrastive cues, sarcasm, or inherent ambiguity, as shown in Figure 1.

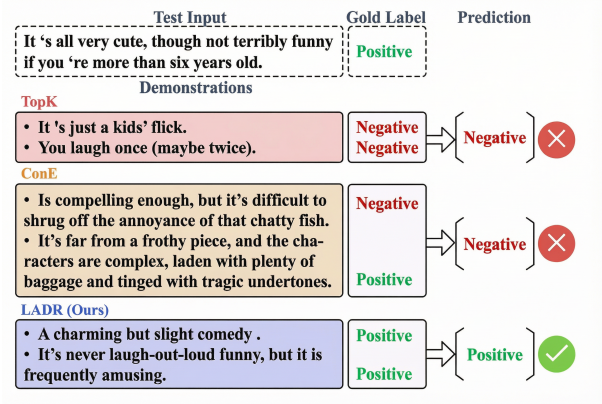


Figure 1: A comparison of 2-shot in-context demonstrations retrieved by different selectors in SST-2.

This problem is amplified in real-world settings with noisy or unreliable labels in the candidate pool. Recent findings further suggest that performance gains from demonstrations may not solely arise from perfectly accurate input-label pairings (Fei et al., 2023; Lu et al., 2024; Zhao et al., 2025); indeed, demonstrations with random (Min et al., 2022) or symbolic labels (Wei et al., 2023) may still work competitively in some cases. Together, these observations indicate that robust demonstration selection should go beyond semantics-only retrieval and explicitly consider label-space compatibility.

To address this gap, we propose **L**abel-**A**lignment **D**ivergence **R**eranking (LADR), a two-stage demonstration selection framework grounded in the principle that helpful demonstrations should be both semantically relevant and label-consistent.

LADR first applies standard semantic retrieval (Liu et al., 2022) to obtain a compact candidate

pool. It then introduces a lightweight, model-agnostic reranking step based on predictive label-distribution alignment.

Concretely, we fine-tune a small language model (SLM), e.g., BERT (Devlin et al., 2019), on the training data to estimate label probability distributions for the test input and each retrieved candidate. We quantify their discrepancy via Jensen-Shannon (JS) divergence (Menéndez et al., 1997) (built upon KL divergence (Kullback and Leibler, 1951)) and rerank candidates to prefer demonstrations with minimal label-distribution divergence from the test input. This label-aware reranking mitigates the failure mode where semantically similar demonstrations carry contradictory labels (e.g., due to negation or sarcasm), improving robustness under noisy demonstration pools (Guo et al., 2021).

LADR is inference efficiency. Unlike feedback-dependent selection methods that require scoring candidates with the target LLM (Wang et al., 2024a; Zhang et al., 2025b), LADR performs reranking using only one SLM forward pass per test input and lightweight divergence computations over the retrieved TopK candidates. Moreover, candidate-side label distributions can be precomputed and cached offline, making the online overhead small.

Our main contributions are:

- We identify a critical limitation of semantics-only demonstration selection for ICL: semantic similarity can conflict with label consistency under ambiguity and label noise.
- We propose LADR, a simple, model-agnostic, and , and interpretable framework that reranks semantically retrieved candidates by JS-based label-distribution alignment estimated by a lightweight SLM.
- We provide a practical offline-online design and comprehensive efficiency evaluation, showing that LADR adds negligible selection overhead while improving ICL accuracy.
- We demonstrate consistent gains across seven benchmarks and multiple LLMs/model scales, and analyze robustness under noisy or out-of-domain demonstration pools.

2 Method

We propose **LADR**, a two-stage demonstration selection framework for in-context text classification. Given a pool of labeled data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ with label set $\mathcal{Y} = \{1, \dots, C\}$, our goal is to select a small set of demonstrations $\mathcal{S}(x) \subset \mathcal{D}$ for each test input x. LADR first retrieves semantically relevant candidates and then reranks these candidates by predictive label-distribution alignment estimated via a lightweight model. An overview is shown in Figure 2.

2.1 Stage I: Semantic retrieval (TopK)

We first retrieve a small candidate set based on semantic similarity. Let $f_{\text{emb}}(\cdot)$ denote a pretrained text encoder that maps an input to an embedding vector. For the test input x and each demonstration input x_i, we compute

$$\mathbf{e}_x = f_{\text{emb}}(x), \qquad \mathbf{e}_i = f_{\text{emb}}(x_i). \tag{1}$$

We use cosine similarity for semantic relevance:

$$S_{\text{text}}(x, x_i) = \cos(\mathbf{e}_x, \mathbf{e}_i) = \frac{\mathbf{e}_x^\top \mathbf{e}_i}{\|\mathbf{e}_x\|_2 \, \|\mathbf{e}_i\|_2}. \tag{2}$$

We then retrieve the TopK most similar demonstrations to form a candidate pool:

$$\mathcal{C}_K(x) = \text{TopK}_{(x_i, y_i) \in \mathcal{D}} \, S_{\text{text}}(x, x_i), \tag{3}$$

where $K \ll N$.

2.2 Stage II: Label-alignment divergence reranking (LADR)

Semantic similarity does not necessarily imply label consistency; therefore, we introduce a probabilistic distribution alignment mechanism. Specifically, we fine-tune a BERT-like model as a lightweight classifier to estimate label distributions.

Predictive label distributions. Let $P_{\text{SLM}}(y \mid x)$ denote the SLM's predicted probability of label $y \in \mathcal{Y}$ for input x. We define the predicted label distribution for the test input as

$$\mathbf{p}_x(y) \triangleq P_{\text{SLM}}(y \mid x), \qquad \mathbf{p}_x \in \Delta^{C-1}, \tag{4}$$

and for each demonstration (x_i, y_i) as

$$\mathbf{p}_i(y) \triangleq P_{\text{SLM}}(y \mid x_i), \qquad \mathbf{p}_i \in \Delta^{C-1}. \tag{5}$$

The SLM is fine-tuned with standard cross-entropy on the task's labeled training data. Importantly, $\mathbf{p}_i$ depends only on x_i and can be precomputed and cached offline for all demonstrations.

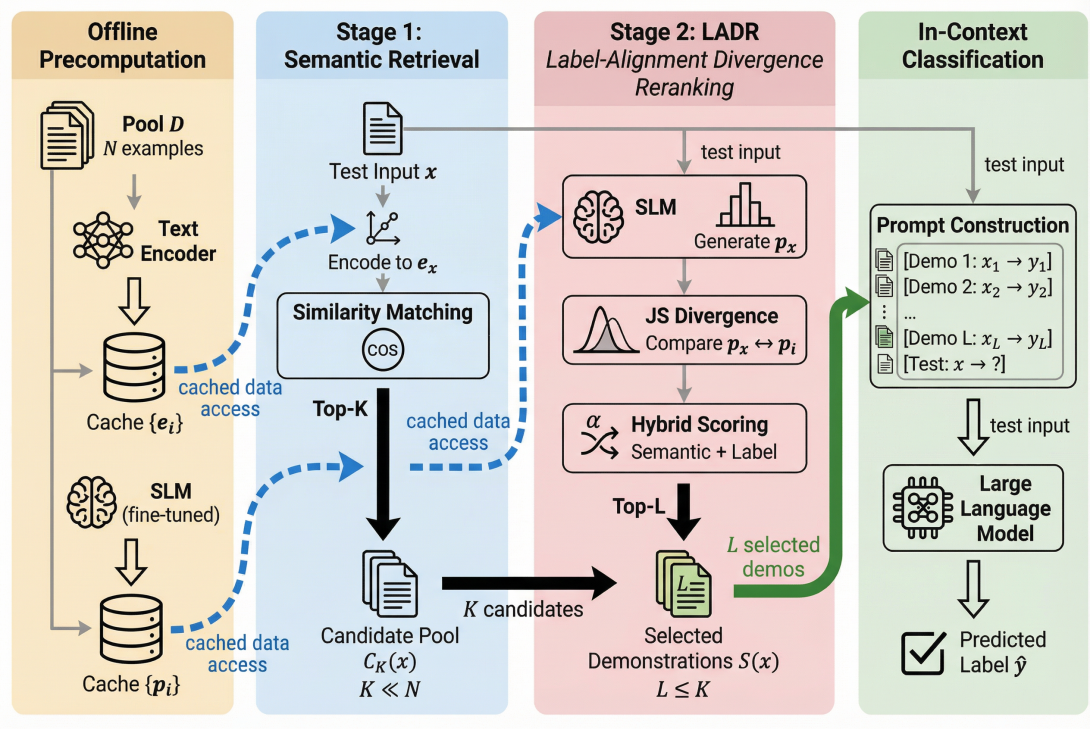


Figure 2: Overview of LADR for demonstration selection and in-context classification.

Distributional discrepancy (JS divergence). To quantify the mismatch between the test distribution $\mathbf{p}_x$ and a candidate distribution $\mathbf{p}_i$, we use Jensen–Shannon (JS) divergence:

$$\mathrm{KL}(\mathbf{p}_x \| \mathbf{m}) = \sum_{y \in \mathcal{Y}} \mathbf{p}_x(y) \log \frac{\mathbf{p}_x(y)}{\mathbf{m}(y)}, \tag{6}$$

$$\mathrm{KL}(\mathbf{p}_i \| \mathbf{m}) = \sum_{y \in \mathcal{Y}} \mathbf{p}_i(y) \log \frac{\mathbf{p}_i(y)}{\mathbf{m}(y)}, \tag{7}$$

$$\mathbf{m} \triangleq \tfrac{1}{2}(\mathbf{p}_x + \mathbf{p}_i), \tag{8}$$

$$\mathrm{JS}(\mathbf{p}_x, \mathbf{p}_i) \triangleq \tfrac{1}{2}\mathrm{KL}(\mathbf{p}_x \| \mathbf{m}) + \tfrac{1}{2}\mathrm{KL}(\mathbf{p}_i \| \mathbf{m}). \tag{9}$$

Unless otherwise stated, we use $\log_2$, in which case $\mathrm{JS}(\mathbf{p}_x, \mathbf{p}_i) \in [0, 1]$.

We define the label matching score as

$$S_{\text{label}}(x, x_i) \triangleq 1 - \mathrm{JS}(\mathbf{p}_x, \mathbf{p}_i), \tag{10}$$

where larger values indicate better agreement in predicted label distributions.

Hybrid reranking score. We combine semantic relevance and label alignment with a convex mixture:

$$S_{\text{hybrid}}(x, x_i) = \alpha \cdot S_{\text{text}}(x, x_i) + (1-\alpha) \cdot S_{\text{label}}(x, x_i), \tag{11}$$

where $\alpha \in [0, 1]$ balances the two signals. We rerank candidates in $\mathcal{C}_K(x)$ by S_{hybrid} and select the top-L demonstrations:

$$\mathcal{S}(x) = \mathrm{TopL}_{(x_i, y_i) \in \mathcal{C}_K(x)} \, S_{\text{hybrid}}(x, x_i), \quad L \le K. \tag{12}$$

2.3 Prompting for in-context classification

Given $\mathcal{S}(x) = \{(x_{(j)}, y_{(j)})\}_{j=1}^{L}$, we construct an ICL prompt by concatenating demonstrations with a fixed template followed by the test input x. The target LLM then predicts a label (e.g., via label verbalizers or constrained decoding over $\mathcal{Y}$).

2.4 Efficiency: Offline–online decomposition

Offline, we precompute and cache (i) demonstration embeddings $\{\mathbf{e}_i\}$ and (ii) predicted label distributions $\{\mathbf{p}_i\}$ for all demonstrations. Online, for each test input we only compute one embedding $\mathbf{e}_x$, one SLM forward pass to obtain $\mathbf{p}_x$, and JS scores for K candidates.

Since JS computation is $O(C)$ per candidate, the reranking overhead is $O(KC)$ per test input, which is negligible compared to per-candidate LLM scoring used by many alternative selection methods. Since JS-based reranking costs only $O(KC)$ per test input (with cached $\mathbf{p}_i$), the additional overhead is lightweight.

Method	AgNews	CR	SST-2	SST-5	Subj	MNLI	QNLI	Average
Random	68.12	92.82	95.22	44.93	73.95	76.41	82.87	76.33
BM25	75.77	93.62	95.00	46.38	90.85	79.08	82.67	80.48
TopK(Liu et al., 2022)	75.71	93.62	96.05	50.23	92.45	80.01	82.76	81.55
TopK + MDL(Wu et al., 2023)	77.83	94.41	96.05	50.05	92.35	79.90	82.76	81.91
TopK + ConE(Peng et al., 2024)	80.95	93.88	95.61	48.91	90.75	78.61	84.26	81.85
MAPLE (Chen et al., 2025)	79.44	94.22	95.00	50.56	69.11	83.67	84.44	79.49
GenICL (Zhang et al., 2025c)	**86.66**	93.35	94.15	51.63	93.35	**85.95**	83.35	84.06
LADR(Ours)	85.76	**94.68**	**96.49**	**54.30**	**95.15**	84.31	**85.45**	**85.16**

Table 1: Different demonstration selection methods with Qwen2.5-7B-Instruct 8-shot performance comparison in accuracy(%) across seven tasks. **Bold** numbers indicate the best performance, while underline values denote the second-best results.

Algorithm 1 LADR demonstration selection

Require: Pool $\mathcal{D}$ with cached $\{\mathbf{e}_i, \mathbf{p}_i\}_{i=1}^{N}$, test input x, hyperparameters K, L, α

1: Retrieve $\mathcal{C}_K(x)$ by TopK cosine similarity using embeddings
2: Compute $\mathbf{p}_x \leftarrow P_{\text{SLM}}(\cdot \mid x)$
3: **for** $(x_i, y_i) \in \mathcal{C}_K(x)$ **do**
4: Compute $S_{\text{text}}(x, x_i)$ and $\text{JS}(\mathbf{p}_x, \mathbf{p}_i)$
5: Compute $S_{\text{hybrid}}(x, x_i)$
6: **end for**
7: Select $\mathcal{S}(x)$ as Top-L by S_{hybrid} and build the ICL prompt

We further confirm this empirically in Figure 3, where LADR yields consistently lower inference time than alternative selection methods under the same experimental conditions.

3 Main Results

We evaluate the effectiveness of the proposed LADR on seven classification tasks. The experimental results (shown in Table 1) demonstrate that:

Our method consistently outperforms other baselines on almost all tasks. LADR achieves a higher overall average than both Maple and GenICL, improving over them by 5.67% and 1.10% on average, respectively, while remaining competitive on the two tasks where GenICL attains the best performance (AgNews and MNLI).

Meanwhile, LADR surpasses strong select-then-rank frameworks such as TopK+ConE and TopK+MDL on six out of seven tasks, achieving an average accuracy gain of 3.31% and 3.25%, respectively, as shown in Table 1.

Compared to the standard TopK retriever, the improvement increases further to 3.61%, highlighting the effectiveness of incorporating label-distribution alignment in the demonstration reranking.

For NLI tasks, LADR obtains the best performance in QNLI with a gain of 1.01% over the second-best method, and in MNLI it still substantially outperforms TopK+ConE and TopK+MDL by 5.70% and 4.41%, respectively, demonstrating its effectiveness in semantically challenging tasks.

In addition, we conduct a paired t-test across 10 seeds comparing LADR, TopK+ConE and TopK+MDL. The results indicate that LADR significantly outperformes TopK+ConE (t=11.6815, p<0.01) and TopK+MDL (t=7.3556, p<0.01).

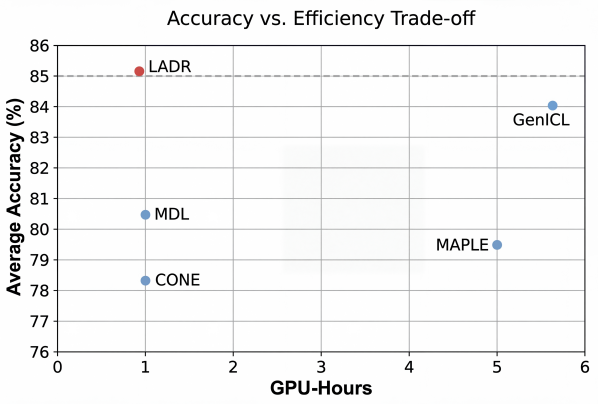


Figure 3: Accuracy–efficiency trade-off between different baselines.

Our method obtains highest performance while retains lowest inference cost among all baselines. We further examine the practical trade-off between classification performance and computational cost for different selection methods. Figure 3 plots the average accuracy (across seven tasks) against the total GPU-hours required by each baselines at the inference time, under the same hardware setup.

Overall, LADR achieves the best accuracy–efficiency balance, residing in the upper-left region of the plot. In particular, LADR attains the

highest average accuracy (85.16%) with less than 1 GPU-hours overhead (0.93 GPU-hours), outperforming GenICL (84.06%, 5.65 GPU-hours) while requiring substantially fewer compute resources (roughly 6× less GPU-hours). MAPLE also incurs a much higher computational cost (around 5 GPU-hours) but yields a noticeably lower average accuracy (79.49%).

These results confirm that LADR not only improves ICL accuracy, but also remains computationally lightweight compared to recent feedback- or pseudo-labeling-based alternatives.

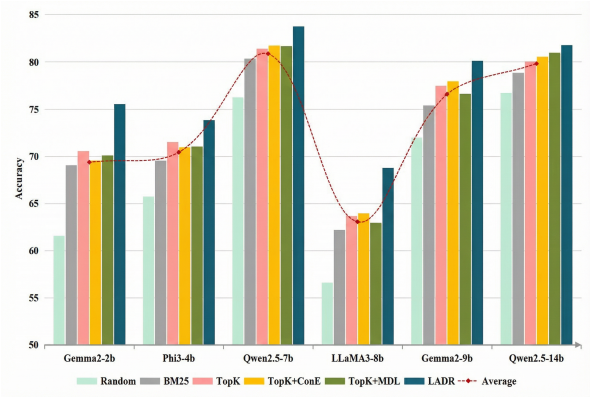


Figure 4: The average accuracy of LLMs on seven tasks at different scales shows that our method consistently improves performance across various models. The red dotted line illustrates the averaged performance of all methods across different model scales.

Our method brings consistent performance improvements across various models and model scales. We evaluate various demonstration selection methods across multiple LLMs (i.e., Gemma2, Phi3, Qwen2.5, and LLaMA3) and model scales ranging from 2B to 14B, as illustrated in Figure 4.

Experimental results demonstrate that our method consistently outperforms baseline approaches across various models and scales, most notably with Gemma2-2B, where it achieves average accuracy gains of 5.53% and 6.08% over TopK + MDL and TopK + ConE, respectively. It is also noteworthy that LLM performance does not always scale positively with model size, as shown by the red dotted line in Figure 4. This observation contrasts with the findings reported in Peng et al. (2024), but aligns with the conclusions drawn in Wang et al. (2023).

4 Analysis

We conduct extensive analyses with Qwen2.5-7B-Instruct on different tasks to further investigate the effectiveness and generalizability of our method.

Our method is robust to arbitrary and reversed labels. Prior studies (Min et al., 2022; Yoo et al., 2022) suggest that the correctness of text–label pairs in in-context demonstrations may have limited influence on ICL performance, as LLMs can partially rely on pattern matching over the inputs or on latent task priors. However, a corrupted candidate pool introduces label-conflicting demonstrations that can increase the variance of the in-context signal and induce unstable predictions, especially when the selected examples are semantically close to the test input.

LADR mitigates this failure mode because reranking is guided by predictive label-distribution alignment rather than the raw label tokens shown in demonstrations. Concretely, LADR compares the SLM-predicted distributions $\mathbf{p}_x$ and $\mathbf{p}_i$, and favors candidates with small $\mathrm{JS}(\mathbf{p}_x, \mathbf{p}_i)$.

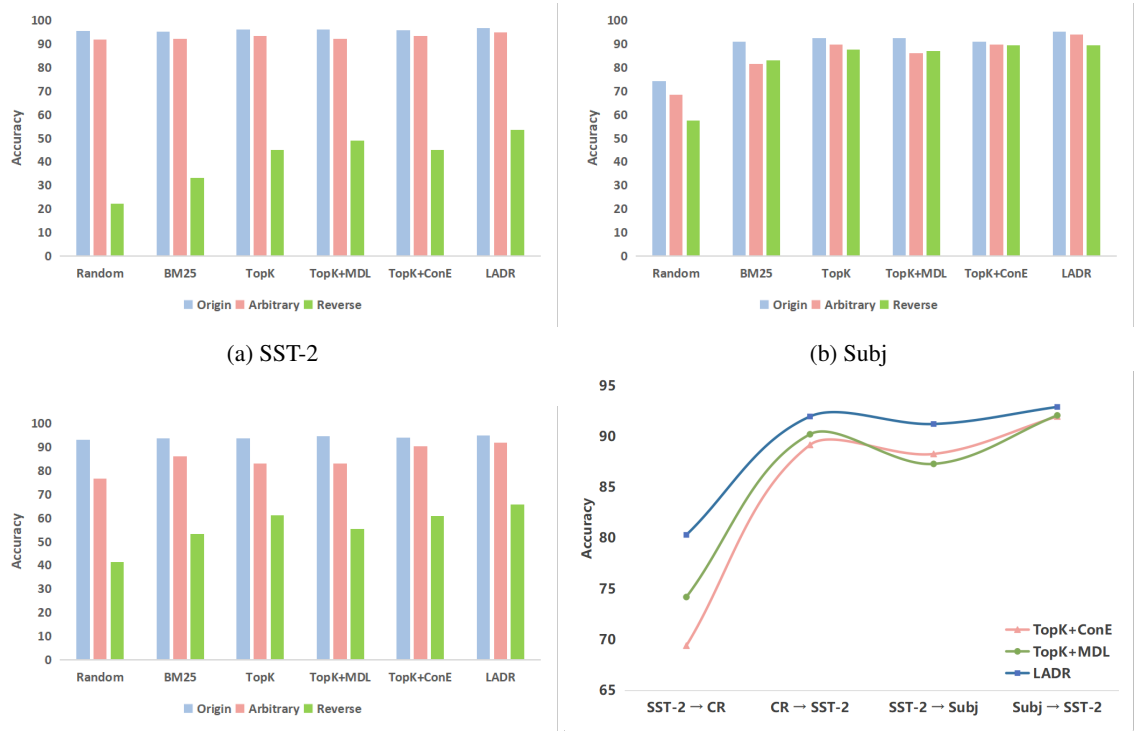
To empirically verify this robustness, we corrupt demonstration labels in two ways: (i) replacing labels with arbitrary symbols (e.g., `foo`/`bar`), and (ii) reversing labels (e.g., flipping `positive` to `negative`).

Figures 5(a)-(c) show that LADR consistently outperforms all baselines under both perturbations. Notably, these results indicate that label correctness can still substantially affect ICL accuracy once demonstration selection is constrained to semantically similar candidates, since semantic retrieval may preferentially surface mislabeled examples that are highly confusable with the test input.

Finally, we observe that most methods are less sensitive to label reversal on Subj than on SST-2 and CR. A plausible explanation lies in the subjective–objective distinction, which remains more nuanced and less tightly coupled to a single polarity cue. By contrast, reversing sentiment labels in SST-2 and CR directly contradicts the dominant semantic intent of the input–label pairs, thereby introducing stronger label conflict.

Our method works for out-of-domain (OOD) demonstration pools. While previous results have demonstrated the effectiveness of our method in in-domain demonstration pools, we now evaluate its generalizability in OOD settings, where demonstrations are drawn from domains different from the target task.

LADR is less sensitive to this shift because it favors demonstrations with small $\mathrm{JS}(\mathbf{p}_x, \mathbf{p}_i)$. This criterion acts as an additional filter that suppresses



(a) SST-2 (b) Subj

(c) CR (d) Out-of-domain

Figure 5: (a) - (c) Performance comparison between original, arbitrary and reverse labels in in-context demonstrations across SST-2, Subj and CR. (d) Performance of our method on out-of-domain demonstration pools. 'A→B' indicates that the demonstration pool is sourced from dataset 'A' while evaluation is conducted on dataset 'B'.

candidates that may be semantically related but induce incompatible label tendencies under domain shift, thereby reducing the effective label conflict introduced by OOD examples.To test this hypothesis, we construct cross-domain demonstration pools among three sentiment-related datasets: Movie Reviews (SST-2), Customer Reviews (CR), and Subjectivity Analysis (Subj).

In each setting, we draw demonstrations from one dataset and evaluate on another, systematically covering domain shifts. As shown in Figure 5(d), LADR consistently outperforms other select-rerank frameworks across all OOD scenarios, with average gains of 3.16% and 4.40% over TopK+MDL and TopK+ConE, respectively. These results suggest that LADR provides a robust selection signal that remains effective even when semantic similarity becomes less reliable under domain mismatch.

5 Impact of hyperparameters

This section conducts a comprehensive analysis of how different hyperparameter settings influence LADR's performance.

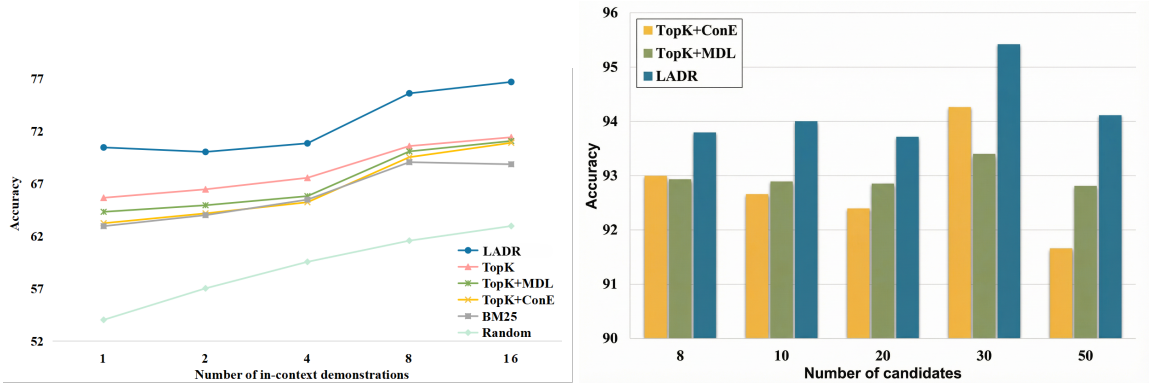
Impact of the number of in-context demonstrations. We begin by examining how the number of in-context demonstrations influences model performance. Specifically, we incrementally increase the number of demonstrations from 1 to 16 and evaluate the results using Gemma2-2B-it. The average accuracy of all methods is calculated and presented in Figure 6(a).

We observe that increasing the number of in-context demonstrations consistently leads to improved performance on average, suggesting a positive correlation between LLM performance and the number of the provided demonstrations.

Notably, LADR consistently outperforms all baseline approaches across various settings. Moreover, as the number of in-context demonstrations increases, the performance gains of our method also grow, achieving significant improvements of 5.48% and 6.11% over TopK+MDL and TopK+ConE, respectively.

Impact of the number of TopK candidates. We examine how varying the number of candidates retrieved from the demonstration pool via the TopK method during the semantic retrieval stage affects performance, using Qwen2.5-7B-Instruct as the evaluation model.

Specifically, we vary the number of candidate



(a) (b)

Figure 6: Performance comparison between different number of (a) in-context demonstrations and (b) candidates in semantic retrieval stage.

Method	CR	SST-2	SST-5	Subj	Average
LADR-BERT-Gemma2	92.55	94.51	34.25	94.90	79.05(+29.35)
LADR-BERT-Phi3	89.89	94.34	42.03	92.04	79.58(+29.88)
BERT	50.80	50.25	24.48	73.25	49.70 (+0)
LADR-RoBERTa-Gemma2	92.55	94.18	35.38	95.20	79.33(+35.50)
LADR-RoBERTa-Phi3	90.69	94.67	41.89	91.79	79.76(+35.93)
RoBERTa	36.17	49.53	18.05	71.55	43.83 (+0)
LADR-DeBERTa-Gemma2	93.09	95.17	37.24	95.60	80.28(+31.42)
LADR-DeBERTa-Phi3	90.96	94.56	42.84	92.39	80.19(+31.33)
DeBERTa	47.07	53.10	13.80	69.45	48.86 (+0)

Table 2: 8-shot performance comparison between standalone SLMs and different SLMs with LADR. (+) indicates the relative improvements achieved when combining our LADR method with various SLMs on Gemma2-2B-it and Phi3-mini-128k-Instruct.

demonstrations retrieved during the semantic retrieval stage from 8 (i.e., equal to the number of in-context demonstrations used) up to 50 across SST-2, Subj and CR datasets, to explore how a larger candidate pool influences the overall performance, as shown in Figure 6(b).

We observe that increasing the number of TopK candidates does not lead to performance improvements until the candidate size reaches 30. However, performance degrades when the candidate size increases to 50. Moreover, a larger candidate set introduces additional latency during the semantic retrieval stage.

We hypothesize that overly large candidate pools introduce more semantically distant "noise" that even a high quality LADR reranker cannot fully eliminate, thereby slightly hurting accuracy and increasing latency. Therefore, we set the default number of TopK candidates to 30, balancing performance gains and computational efficiency.

Notably, our method consistently outperforms other baselines across all candidate settings, demonstrating its robustness and effectiveness with extended demonstration candidates.

Impact of small language models. LADR measures candidate–input compatibility using SLM-predicted label distributions. To isolate this effect, we vary the SLM while keeping the rest of the pipeline fixed (8-shot setting).

We consider three encoder-only Transformer classifiers, BERT-base-uncased, RoBERTa-base, and DeBERTa-v3-base, and evaluate both their standalone performance and the corresponding LADR variants with two LLMs on four datasets. As shown in Table 2, all three SLMs are relatively weak standalone classifiers, yet once combined with LADR and an LLM, their averages jump into the 79–80% range. For example, LADR-BERT-Gemma2 improves over BERT alone by +29.35 points on average, and LADR-RoBERTa-Gemma2 and LADR-RoBERTa-Phi3 yield even larger gains of +35.50 and +35.93 points over RoBERTa, respectively. DeBERTa-based variants also show strong boosts (+31.42 and +31.33 over DeBERTa).

Across SLMs, DeBERTa-based LADR consistently attains the best average performance (80.28 with Gemma2 and 80.19 with Phi3), with RoBERTa-based LADR slightly outperforming

BERT-based LADR (79.33 vs. 79.05 for Gemma2, 79.76 vs. 79.58 for Phi3). These differences are modest but stable across two distinct LLM families, suggesting that stronger SLMs can provide more informative label-distribution estimates for divergence-based reranking, whereas the majority of the overall accuracy is still governed by the LLM itself. At the same time, the large relative gains over all three SLMs indicate that LADR does not require a highly accurate classifier: even SLMs with moderate global accuracy can supply sufficiently structured posterior distributions for ranking semantically retrieved candidates.

Impact of tunable weight α. LADR combines semantic relevance and label-distribution alignment through a convex mixture controlled by α in Eq. 11. Specifically, α governs the trade-off between a relevance prior $S_{\text{text}}(x, x_i)$, which constrains candidates to be topically related to the test input, and a label-alignment term $S_{\text{label}}(x, x_i)$, which promotes label consistency by minimizing the divergence between the predicted label distributions $\mathbf{p}_x$ and $\mathbf{p}_i$.

In particular, when the SLM produces a peaked distribution $\mathbf{p}_x$ (i.e., high confidence), small $\text{JS}(\mathbf{p}_x, \mathbf{p}_i)$ more reliably implies label agreement, so emphasizing $S_{\text{label}}(x, x_i)$ helps avoid label-conflicting demonstrations. Conversely, when $\mathbf{p}_x$ is uncertain, over-weighting $S_{\text{label}}(x, x_i)$ may amplify estimation noise, and $S_{\text{text}}(x, x_i)$ serves as a robust fallback. Therefore, an intermediate α is expected to be optimal in practice, reflecting a bias–variance trade-off between semantic retrieval and distributional alignment.

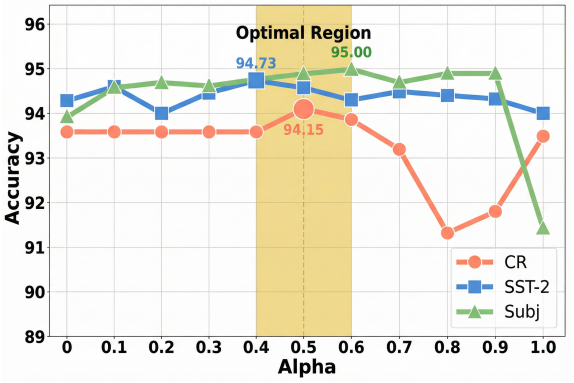


Figure 7: Performance comparison of different α settings.

To empirically study this effect, we vary α from 0 to 1 in increments of 0.1 using Qwen2.5-7B-Instruct and report the results in Figure 7. We observe that accuracy is consistently maximized when α lies in $[0.4, 0.6]$, suggesting that LADR performs best when semantic relevance and label-distribution consistency are properly balanced. Accordingly, we set $\alpha = 0.5$ as the default value in all experiments.

Ablation studies. We further conduct ablations using two extreme settings. When $\alpha = 1$ (w/o Align), LADR reduces to a pure semantics-only selector, i.e., ranking $\mathcal{C}_K(x)$ solely by $S_{\text{text}}(x, x_i)$. When $\alpha = 0$ (w/o Sem), we still retrieve the same candidate pool $\mathcal{C}_K(x)$ via TopK, but rerank candidates solely by $S_{\text{label}}(x, x_i)$, removing semantic ranking. These ablations quantify the individual contributions of semantic relevance and distributional alignment, and confirm that neither signal alone is sufficient to reach the best performance.

Method	CR	SST-2	Subj
LADR	94.68	96.49	95.15
w/o Sem	93.62(↓1.06)	94.31(↓2.18)	93.90(↓1.25)
w/o Align	93.62(↓1.06)	96.05(↓0.44)	92.45(↓2.70)

Table 3: Ablation study on three datasets. **w/o Sem** refers to a setting in which demonstrations are ranked solely based on label distributional consistency, omitting semantic similarity scoring. **w/o Align** denotes the vanilla top-K selection method without the application of LADR.

We observe that the performance of LADR degrades proportionally when each component is removed. Specifically, the average performance drops by 1.50% when semantic sorting is excluded, and by 2.30% when the label distributional consistency score is removed. These results highlight that both components are essential and jointly contribute to the optimal performance of our method.

6 Conclusion

In this paper, we propose **LADR**, a two-stage demonstration selection framework designed to improve in-context learning for text classification. LADR addresses a key limitation of semantics-only demonstration selection: under label ambiguity or noisy demonstration pools, semantic similarity alone can retrieve label-conflicting examples. By incorporating label-distribution alignment into the selection process, LADR promotes label-consistent demonstrations while preserving semantic relevance, leading to more reliable ICL performance.

Limitations

We acknowledge several limitations in our work. (1) The selection of LLMs is constrained to model scales between 2B and 14B and to a limited set of instruction-tuned families due to computational constraints; the extent to which our findings transfer to larger or fundamentally different LLM architectures remains to be explored. (2) Our method relies on a supervised SLM to estimate label distributions. This requires access to labeled data for fine-tuning and restricts our study to encoder-only Transformer classifiers (BERT, RoBERTa, DeBERTa). Although even moderately strong SLMs already yield substantial gains with LADR, the impact of alternative SLM architectures and training paradigms is not investigated here. (3) While we have focused on the effect of label-distribution alignment in the reranking stage, the overall performance still depends on the underlying TopK semantic retrieval. We do not vary the retriever or jointly optimize retrieval and reranking; a more systematic study of different retrieval backbones and their interaction with label-distribution alignment is left for future work.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, and 1 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

AI@Meta. 2024. Llama 3 model card.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zijie Cai, Hui Fang, Jianhua Liu, Ge Xu, Yunfei Long, Yin Guan, and Tianci Ke. 2025. Improving unified information extraction in chinese mental health domain with instruction-tuned llms and type-verification component. *Artificial Intelligence in Medicine*, 162:103087.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Xiaowen Ding, Bing Liu, and Philip S Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining*, pages 231–240.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, and 1 others. 2024. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128.

Yu Fei, Yifan Hou, Zeming Chen, and Antoine Bosselut. 2023. Mitigating label biases for in-context learning. In *Proceedings Of The 61St Annual Meeting Of The Association For Computational Linguistics (Acl 2023): Long Papers, Vol 1*, pages 14014–14031. Assoc Computational Linguistics-Acl.

Yanjun Gao, Ruizhe Li, Emma Croxford, John Caskey, Brian W Patterson, Matthew Churpek, Timothy Miller, Dmitriy Dligach, and Majid Afshar. 2025. Leveraging medical knowledge graphs into large language models for diagnosis prediction: Design and application study. *JMIR AI*, 4:e58670.

Gemma. 2024. Gemma.

Biyang Guo, Songqiao Han, Xiao Han, Hailiang Huang, and Ting Lu. 2021. Label confusion learning to enhance text classification models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 12929–12936.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

SU Hongjin, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and 1 others. 2022. Selective annotation makes language models better few-shot learners. In *The Eleventh International Conference on Learning Representations*.

Dan Iter, Reid Pryzant, Ruochen Xu, Shuohang Wang, Yang Liu, Yichong Xu, and Chenguang Zhu. 2023. In-context demonstration selection with cross entropy difference. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1150–1162.

Ye Jiang and Yimin Wang. 2025. Imfnd: In-context multimodal fake news detection with large visual-language models. *Knowledge-Based Systems*, page 113880.

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *Preprint*, arXiv:2308.03281.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, William B Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for gpt-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Qiang Lu, Yunfei Long, Xia Sun, Jun Feng, and Hao Zhang. 2024. Fact-sentiment incongruity combination network for multimodal sarcasm detection. *Information Fusion*, 104:102203.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8086–8098.

María Luisa Menéndez, Julio Angel Pardo, Leandro Pardo, and María del C Pardo. 1997. The jensen-shannon divergence. *Journal of the Franklin Institute*, 334(2):307–318.

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*, pages 11048–11064.

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

Qwen. 2024. Qwen2.5: A party of foundation models.

Stephen Robertson, Hugo Zaragoza, and 1 others. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

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

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 *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics.

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

Liang Wang, Nan Yang, and Furu Wei. 2024a. Learning to retrieve in-context examples for large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1752–1767.

Shuai Wang, Liang Ding, Li Shen, Yong Luo, Bo Du, and Dacheng Tao. 2024b. Oop: Object-oriented programming evaluation benchmark for large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 13619–13639.

Jerry Wei, Le Hou, Andrew Lampinen, Xiangning Chen, Da Huang, Yi Tay, Xinyun Chen, Yifeng Lu, Denny Zhou, Tengyu Ma, and 1 others. 2023. Symbol tuning improves in-context learning in language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 968–979.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122.

Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. 2023. Self-adaptive in-context learning: An information compression perspective for in-context example selection and ordering. In *The 61st Annual Meeting of the Association for Computational Linguistics (09/07/2023-14/07/2023, Toronto, Canada)*.

Canwen Xu, Yichong Xu, Shuohang Wang, Yang Liu, Chenguang Zhu, and Julian McAuley. 2024. Small models are valuable plug-ins for large language models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 283–294.

Kang Min Yoo, Junyeob Kim, Hyuhng Joon Kim, Hyunsoo Cho, Hwiyeol Jo, Sang-Woo Lee, Sang-goo Lee, and Taeuk Kim. 2022. Ground-truth labels matter: A deeper look into input-label demonstrations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2422–2437.

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

Yuqi Zhang, Liang Ding, Lefei Zhang, and Dacheng Tao. 2025a. Intention analysis makes llms a good jailbreak defender. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2947–2968.

Zheng Zhang, Shaocheng Lan, Lei Song, Jiang Bian, Yexin Li, and Kan Ren. 2025b. Learning to select in-context demonstration preferred by large language model.

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

Qingqing Zhao, Yuhan Xia, Yunfei Long, Ge Xu, and Jia Wang. 2025. Leveraging sensory knowledge into text-to-text transfer transformer for enhanced emotion analysis. *Information Processing & Management*, 62(1):103876.

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2024. Can large language models transform computational social science? *Computational Linguistics*, 50(1):237–291.

A Related Work

Large language models (LLMs) have demonstrated strong performance across diverse NLP tasks (Gao et al., 2025; Cai et al., 2025), and recent studies have examined how prompting and zero-shot/ICL strategies affect their behaviors (Zhang et al., 2025a; Ziems et al., 2024; Wang et al., 2024b, 2023). Despite these advances, ICL is often unstable: minor changes in the selected demonstrations (or their ordering) can lead to large performance fluctuations (Lu et al., 2022; Min et al., 2022), motivating research on demonstration selection.

A dominant line of work selects demonstrations by similarity, e.g., retrieving semantically close examples for each test input (Liu et al., 2022). Beyond pure retrieval, some methods improve the candidate pool via selective annotation and voting schemes (Hongjin et al., 2022), or adopt select–then–rank frameworks with alternative objectives such as Minimum Description Length (TopK+MDL) (Wu et al., 2023), cross-entropy difference signals from a small model (Iter et al., 2023), and conditional-entropy-based reranking (TopK+ConE) (Peng et al., 2024).

More recent learning-based selectors (e.g., GenICL) and many-shot pipelines (e.g., MAPLE) further explore using additional supervision signals (LLM feedback or pseudo-labeling) to optimize selection, but these approaches can introduce extra computational and optimization complexity.

Meanwhile, several studies question whether correct input–label pairings are always necessary for ICL: even randomly assigned labels (Min et al., 2022) or symbolic labels (Wei et al., 2023) can yield competitive results in certain settings, and recent analyses suggest that gains from demonstrations may not primarily stem from perfectly correct pairings (Fei et al., 2023; Lu et al., 2024; Zhao et al., 2025).

Dataset	Train	Val	Test	Labels	Task
SST-2(Socher et al., 2013)	6,920	872	1,821	2	Sentiment Classification
SST-5(Socher et al., 2013)	8,544	1,101	2,210	5	Sentiment Classification
CR(Ding et al., 2008)	3,394	0	376	2	Sentiment Classification
Subj(Wang et al., 2018)	8,000	0	2,000	2	Subjectivity Analysis
AgNews(Zhang et al., 2015)	120,000	0	7600	4	Topic Classification
MNLI(Williams et al., 2018)	392,702	19,647	19,643	3	Natural Language Inference
QNLI(Wang et al., 2018)	104,743	5,463	5,463	2	Natural Language Inference

Table 4: The statistics of datasets.

B Experimental Setup

Datasets. We conduct a comprehensive evaluation across seven text classiffcation tasks for evaluating the generalizability of the proposed LADR, including: SST-2 (Socher et al., 2013), CR (Ding et al., 2008), Subj (Wang et al., 2018), SST-5 (Socher et al., 2013), AgNews (Zhang et al., 2015)), MNLI (Williams et al., 2018) and QNLI (Wang et al., 2018).

The details of datasets used in our experiments are provided in Table 4. All datasets are sourced from the HuggingFace Hub. For most datasets, we report results on the official test sets. However, for MNLI and QNLI, we report results on their validation sets due to restricted access to the corresponding test sets.

Large language models. We evaluate our method across a range of LLMs, employing `Qwen2.5-7B-Instruct` (Qwen, 2024) as the primary model for most experiments. To assess the scalability and generalizability of the LADR method, we further conduct experiments using LLMs of varying sizes, from 2B to 14B parameters.

These include `Gemma2-2B-it` (Gemma, 2024), `Phi3-mini-128k-Instruct` (Abdin et al., 2024), `LLaMA3-8B-Instruct` (AI@Meta, 2024), `Gemma2-9B-it` (Gemma, 2024), and `Qwen2.5-14B-Instruct` (Qwen, 2024).

Small language models. For label distribution estimation, we adopt `RoBERTa-base` (Liu et al., 2019) as the primary model. To assess the influence of SLMs on the performance of LADR, we additionally incorporate two classic SLMs: `BERT-base-uncased` (Devlin et al., 2019) and `DeBERTa-v3-base` (He et al., 2021).

Baselines. We primarily compare our method against seven baselines for in-context demonstration selection.

- **Random** denotes that the in-context demonstrations are randomly selected, and then are inferred by a LLM.
- **BM25** (Robertson et al., 2009) computes word-overlap similarity between training samples and the test input, selecting the most similar samples as demonstrations.
- **TopK** (Liu et al., 2022) selects the nearest neighbors from training samples for a given test input as its corresponding in-context demonstrations.
- **TopK + MDL** (Wu et al., 2023) adopt a select-then-rank framework, in which demonstrations retrieved via the TopK method are ranked according to the Minimum Description Length (MDL) principle.
- **TopK + ConE** (Peng et al., 2024) is a data- and model-dependent demonstration selection method, which posits that effective demonstrations are those that minimize the conditional entropy of the test input under the inference model.
- **MAPLE** (Chen et al., 2025) is a many-shot ICL framework that leverages unlabeled data via adaptive pseudo-labeling. It first identifies the most impactful unlabeled samples, queries an LLM to assign pseudo-labels, and then adaptively selects a large set of (pseudo-)labeled demonstrations tailored to each test input.
- **GenICL** (Zhang et al., 2025c) learns a demonstration selector using LLM feedback. It formulates demonstration selection as a generative preference learning problem and directly optimizes for demonstrations that are preferred by the target LLM, rather than relying solely on surrogate objectives such as semantic similarity.

Experimental Details. To ensure a fair comparison, we follow the experimental setup introduced by Peng et al. (2024). All experiments are conducted on two RTX 4090 GPUs, with a fixed random seed to ensure reproducibility.

Specifically, we use the TopK method to retrieve 30 candidate demonstrations for each test sample, which are then re-ranked using our proposed LADR. Prompt templates are adopted from Lu et al. (2022) and Wu et al. (2023), with full details listed in Appendix C.

We introduce a tunable weight α to balance the semantic similarity score and the label distribution consistency score. Based on empirical validation, α is fixed at 0.5, the detail is illustrated in Figure 7.

The main comparative experiments are conducted under an 8-shot ICL setting, using `gte-base-en-v1.5` (Li et al., 2023) as the default model for semantic retrieval. For label distribution estimation, we use RoBERTa-base as the default SLM. The training set is split in an 8:2 ratio for training and validation. Fine-tuning is performed using the Huggingface Trainer API to evaluate SLM performance.

C Example templates for different tasks

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

Table 5: Prompt templates of all tasks. “<X>” and “<C>” are placeholders for real inputs.