Logit Separability-Driven Samples and Multiple Class-Related Words Selection for Advancing In-Context Learning

Anonymous ACL submission

Abstract

Effective organization of in-context learning (ICL) demonstrations is key to improving the quality of large language model (LLM) responses. To create better sample-label pairs that instruct LLM understanding, we introduce logit separability, a criterion to assess the clarity of both samples and class-related words at the logit level. This facilitates the optimization of sample and label selection, enhancing the precision of information provided in ICL demonstrations. Additionally, we find that incorporating multiple class-related words for each sample, rather than relying on a single class name, improves performance by offering a broader range of label information. Building on these insights, we propose LICL, a logit separability-based method that jointly organizes samples and integrates multiple class-related words into each sample-label pair. Evaluations across seven classification datasets show that this approach significantly improves ICL performance by providing clearer instructions and richer label information.

1 Introduction

In-context learning (ICL) enables large language models (LLMs) to perform new tasks using sample-label pairs as demonstrations, without the need for retraining or fine-tuning (Brown et al., 2020). However, the organization of these sample-label pairs is critical, as it can significantly affect ICL performance[1] (Liu et al., 2022).

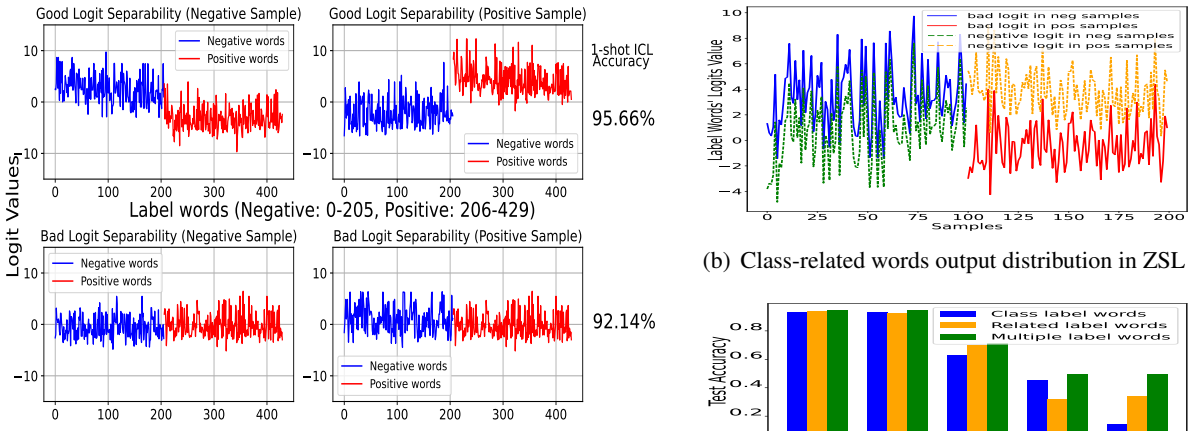
How do we assess the suitability of samples and labels for demonstration? In traditional machine learning, effective features are those that exhibit high discriminative power, enabling clear distinction between their respective classes (Fukunaga, 2013). Analogously, we introduce **logit separability**, a criterion to assess how well in-demonstration samples and label words differentiate between classes at the logit level. Logit separability captures two aspects: (1) how significantly a sample is predicted to its true label, where the correct label has substantially higher logit than others, and (2) how consistently a class-related word yields high logit values across samples of the same label. This measure provides insight into both the separability of samples through class-related words and the ability of these words to distinguish between samples of different classes.

Building on this concept, we observe that not all samples or class-related words exhibit the same degree of logit separability. As shown in Fig.1(a), some samples produce strong separation in the logits of class-related words, assigning high logits to words matching the true label and low logits to others, while others show marginal differences between them. Additionally, certain class-related words consistently display higher logit in samples of their corresponding label compared to samples of other labels. As shown in Fig.1(b), "bad" shows a clearer differentiation between positive and negative samples than the class name "negative".

Given the importance of providing clear label-instructed information in demonstrations, leveraging logit separability for selecting samples and labels might enhance ICL performance. As shown in Fig.1(a)-accuracy, samples with strong logit separability provide more precise signals by aligning class-related words with the model's internal representations. Besides, class-related words with distinct logit separability can beat predefined class names in guiding the model toward accurate predictions (Appx.A.2.1). Therefore, selecting both samples and class-related words via logit separability ensures clearer, more effective demonstrations, ultimately improving the performance of ICL.

Besides, the variability observed in logit values of class-related words across different samples shows that relying on a single class-related

[1] In this paper, we follow Wu et al. (2023) to denote the selection and ranking of sample-label pairs as organization.



(a) Samples' logit separability in ZSL

(b) Class-related words output distribution in ZSL

(c) Multiple class-related words effectiveness in ICL

Figure 1: Exploration of Samples and class-related words on LLaMA2-7b. (a): Logit separability of SST-2 samples across class-related words under zero-shot learning (ZSL), showing varying degrees of separation due to different input samples. The 1-shot accuracy is demonstrated using the good or bad negative/positive samples, with class names as labels. (b): Logit values of various class-related words for 100 negative and 100 positive SST-2 samples under ZSL, showcasing the logit separability of class-related words across samples. (c): Accuracy comparison in 1-shot ICL using class names, single class-related words, and multiple class-related words (combining the two sets with spaces) as labels. Performance with multiple class-related words surpassed the other two sets. More experiments and analyses, including those on GPT2-xl, are in Appx.A.

word may not adequately capture the full semantics associated with a class name. Unlike expanding candidates via a verbalizer in prediction mapping, directly inserting class-related words into demonstrations offers explicit instructions that guide LLM responses to potential candidates. Consequently, transitioning from single to multiple class-related words could enhance linguistic precision and deepen the semantic representation of class names, potentially leading to enhanced model performance. Fig.1(c) supports that using multiple class-related words in sample-label pairs indeed strengthens ICL performance.

Given samples and labels' critical roles in demonstration effectiveness for ICL, in this work, we propose a logit separability-based demonstration organization method. We organize demonstration samples based on their zero-shot logit separability across a pool of class-related words. In demonstration label selection, we propose a novel method that utilizes multiple class-related words instead of a single one. The selection, ordering, and quantity of class-related words are determined through sequential forward search, guided by logit feedback from the selected samples and their validation performance. Moreover, while most research investigates either the sample selection (Zhang et al., 2022; Hongjin et al., 2022; Levy et al., 2023; Lu et al., 2022) or label selection (Yoo et al., 2022; Milios et al., 2023; Gao et al., 2023; Peskine et al., 2023) individually, we comprehensively address both aspects to enhance ICL.

We summarize our contributions as follows:

1. We introduce **Logit Separability**, a criterion that evaluates how clearly a sample or label distinguishes between classes at the logit level, helping assess the suitability of samples and labels for demonstrations in ICL.
2. We propose **LICL**, a **L**ogit separability-based demonstration organization method for **ICL**. We strategically select and order both samples and labels via the class-related words' logit separability in LLM's output space, thereby improving the demonstration's instruction efficacy.
3. We present a novel label instruction method for ICL that inserts multiple class-related words in sample-label pairs. This approach provides richer label information, improving breadth, reducing ambiguity, and enhancing ICL performance.

2 Related Work

Demonstration organization in ICL To improve ICL performance with better demonstration organization, some studies use pre-trained models like S-BERT (Liu et al., 2022) or BM25 (Hongjin et al., 2022; Levy et al., 2023) to select and rank demonstrations. While these unsupervised methods have advantages, they may cause inconsistencies

in knowledge transfer. Other approaches organize demonstration based on the LM's output distribution. Some methods take part of the training set as validation to enable supervised learning methods for demonstration organization (Chang and Jia, 2023; Zhang et al., 2022; Wang et al., 2024). However, this will shrink the pool of candidates, risking sub-optimal selection. Additionally, some methods use the LM's output, like label confidence, to organize demonstrations under the full training set (Rubin et al., 2022; Wu et al., 2023; Li and Qiu, 2023). In the above works, the class names are preassumed to be the label word when creating the demonstration sample-label pairs.

Label matters in ICL The significance of demonstration labels in ICL has been debated. Yoo et al. (2022) demonstrates a positive relationship between ICL performance and accurate sample-label mapping. Li and Qiu (2023) reveals that using the same sample but different class-related words as labels in the demonstrations can result in very different ICL performance. Wang et al. (2023) suggests that labels derive semantic representations from demonstrations for use in deep layers to make final predictions. Yu and Ananiadou (2024) further shows that these demonstration features are integrated into corresponding labels, with each in-context head extracting features specific to these labels. Milios et al. (2023) uses different class-related words in augmented samples to enhance ICL performance, indicating the potential for enhanced demonstration effectiveness when multiple class-related words are used. However, this approach significantly increases the required token length and running time in augmentation settings.

3 Method

In this section, we introduce our proposed method **LICL**, which comprises logit separability-based demonstrated sample selection and ordering, and corresponding multiple class-related words insertion, as shown in Fig.2.

3.1 Problem Statement

Give a large language model M, label space (class names) L, a class-related word pool P with class-related words (includes class names), test sample x_{test} and demonstrations $\{x_i, y_i\}_{i=1}^{L}$[2]. The zero-shot classification of x_{test} can be based on the logit of class names as $\arg\max_{y \in L} p_M(y \mid x_{\text{test}})$ or based on the logit of all class-related words in the pool as $\arg\max_{y \in P} p_M(y \mid x_{\text{test}})$. Similarly, the 1-shot prediction of x_{test} is $\arg\max_{y \in L} p_M(y \mid x_1 \oplus y_1, \cdots, x_L \oplus y_L \oplus x_{\text{test}})$ or $\arg\max_{y \in P} p_M(y \mid x_1 \oplus y_1, \cdots, x_L \oplus y_L \oplus x_{\text{test}})$. $\oplus$ is the concatenation operation, and function $p_M(\cdot)$ returns the logit of words in M's vocabulary.

3.2 Logit Separability-Based Sample Selection and Ordering

A well-chosen sample should exhibit strong logit separability, producing higher logits for class-related words that correspond to the sample's label, with these words ranked as highly as possible among all words in the pool P based on logit values. This ensures that the selected samples are aligned with multiple correlated class-related words while minimizing alignment with uncorrelated words, enhancing the sample's class-related representativeness and improving the effectiveness of ICL demonstrations.

Pool Refinement The initial pool P is constructed from existing works (Hu et al., 2022; Zhu et al., 2024) with details in Appx.D. However, not all words in P exhibit correct separable features within both the dataset and the LLM, risking the introduction of incorrect class-related features. To refine P and maintain the quality of the logit separability measurement, we propose a filtering method to generate a refined pool, P_r, by selecting class-related words based on their logit separability and correlation using point-biserial testing (Kornbrot, 2014) across dataset samples. Details of this refinement are in Appx.B, with analyses in Appx.H.1.

With the refined pool P_r, for each zero-shot learning sample, the LLM generates logit values for all words in P_r. We first remove samples where the class-related word with the highest logit corresponds to an incorrect label. Next, we score samples for selection. Since predictions only rely on the words with the highest logit, scoring based on all words introduces noise—particularly in multi-class tasks—because the pool contains far more words linked to incorrect labels than to the correct label. Hence, we focus exclusively on the true label pool, using two methods—Top-Logit Summation and Rank-Weighted Counting. These methods ensure that selected samples align with logit separability across relevant labels, yielding higher logits for class-related words matching the sample's label and ranking them among the top in P_r. Let N_l be

[2] Our work focus on 1-shot ICL with label-balanced demonstrations, so labels in demonstration are from the 1-th to the L-th label. Discussion of label balance can be found in Appx.H.4

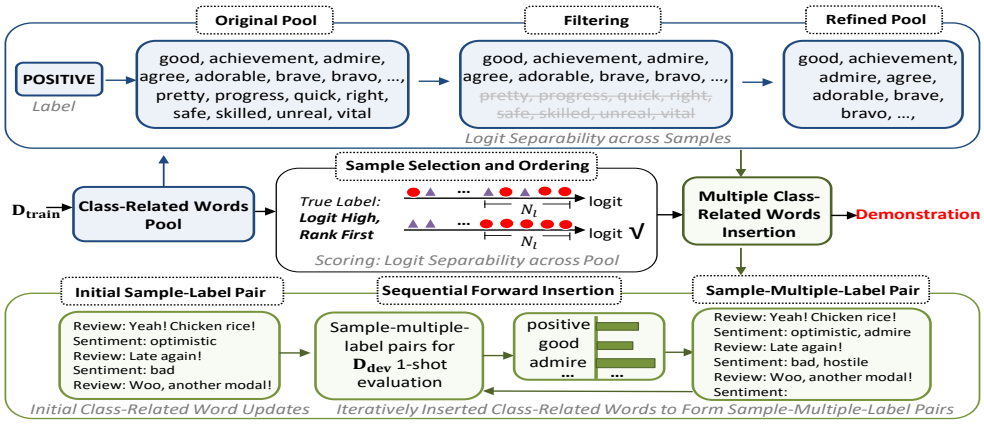


Figure 2: Overall architecture of LICL: The top part shows the pool refinement, with sample organization based on logit separability across the refined pool. The bottom part presents multiple class-related word insertion via sequential forward search, starting from an initial sample-label pair to form a sample-multiple-label pair.

the number of words in P_r for class l, and P_r^l be the pool of class-related words for class l.

Top-Logit Summation For each training sample t_j with label l_{t_j}, we compute its score by summing the logits of the top N_l words from P_r, sorted in descending order by their logits b_{w_i} (Eq.1). Ideally, these top N_l words are all class-related words corresponding to l_{t_j}, indicating strong logit separability. Therefore, only the top N_l words w_i, which belong to the sample's label l_{t_j} (i.e., in $P_r^{l_{t_j}}$), are included in the sum, ensuring that the score reflects only the contribution from the correct class-related words.

$$\text{score}_{t_j} = \sum_{i=0}^{N_l-1} b_{w_i}, \text{ if } w_i \in P_r^{l_{t_j}} \quad (1)$$

Rank-Weighted Counting When the logit values of class-related words are negative, summing logit becomes less effective. To address this, we focus on the ranks of the class-related words rather than their logit values to capture logit separability when logit is negative. Each training sample t_j with label l_{t_j}, is evaluated by assigning linear weights to its corresponding class-related words based on their ranks within the top N_l words from P_r, sorted by their logits b_{w_i} (Eq.2). Higher-ranked words receive higher weights, emphasizing their prominence.

$$\text{score}_{t_j} = \sum_{i=0}^{N_l-1} \frac{2(N_l - i)}{(N_l + 1)N_l}, \text{ if } w_i \in P_r^{l_{t_j}} \quad (2)$$

In k-shot ICL, for each class l, we select the top-k highest-scoring training samples labeled as l. We then construct the demonstration sequence by interleaving these samples across classes: first including the highest-scoring sample from each class, followed by the second-highest, and so on. This ordering ensures that the most logit-separable and representative samples appear early while maintaining balanced class representation, thereby reducing potential bias and enhancing the effectiveness of the demonstrations.

3.3 Multiple Class-Related Words Insertion

As analyzed in Sec.1, relying solely on class names in the sample-label pair may not provide sufficient semantic information for the labels. To address this, we aim to enrich the semantic diversity of label prompts by employing multiple class-related words, proposing the use of a sequence of class-related words to create a sample-multiple-label pair, thereby enhancing the effectiveness of demonstration labels.

After confirming that the selected samples respond well to the multiple class-related words in P_r, our method sequentially inserts these words into a single sample-label pair to form a sample-multiple-label pair, which is more computationally and memory efficient than using multiple augmented sample-label pairs in ICL instruction.

The word insertion process is based on sequential forward search, where the sample-multiple-label pairs are evaluated as 1-shot demonstrations on the validation set. New class-related words are inserted based on the output logit values. Intuitively, since no demonstration is in place, the labels in the initial sample-label pairs typically use predefined class names. However, we find that predefined class names are not always optimal, as they might exhibit lower logit in zero-shot learning compared to other class-related words. Consequently,

we construct the initial sample-label pairs by replacing weak logit response class names with class-related words that yield higher logit values within the selected sample set.

Initial Class-Related Word Updates If a class name l ranks below the top 1 based on its logit score in zero-shot learning across the selected samples from P_r, we replace it with the word that has the highest logit value within the selected sample set. This joint process of assigning initial class-related words from the selected samples ensures that the resulting sample-label pairs are better tailored for the LLM's logit feedback.

Sequential Forward Insertion: After obtaining the initial sample-label pairs, we select and order class-related words to form sample-multiple-label pairs. For each label $l \in L$, we insert the word w_i^l with the highest average logit value across all samples in D_{dev}^l under ICL as Eq.3:

$$w_i^l = \arg\max_{w \in P_r^l} \frac{1}{|D_{dev}^l|} \sum_{x \in D_{dev}^l} b_w^x \qquad (3)$$

where D_{dev} is the validation set derived from the training set with the demonstration samples removed, D_{dev}^l is the subset of D_{dev} containing all samples of class l, and b_w^x is the logit value of word w for validation sample x. After each insertion, the selected word is removed from P_r^l to prevent repeated insertions.

This insertion is iterative, with the selected word updating sample-multiple-label pairs in each round. The iteration continues until no more candidates remain in P_r^l, or until further insertions reduce validation set performance. Examples of prompts for zero-shot, 1-shot, and 1-shot multiple class-related word learning are provided in Appx.C.

4 Experiments

In this section, we examine the capacities of our method in ICL from the following perspectives: (1) 1-shot ICL classification comparison between baseline models and LICL (Sec.4.2); (2) 5-shot ICL (Sec.4.3); (3) Ablation studies on sample ordering (Sec.4.4.1) and multiple class-related words selection and ordering (Sec.4.4.2); (4) Scalability to larger LLMs (Sec.4.5); (5) General applicability of selected multiple class-related words (Sec.4.6); and (6) Further analyses (Sec.4.7).

4.1 Setups

Datasets Seven datasets are evaluated, including: SST-2 (Socher et al., 2013), CR (Ding et al., 2008), IMDB (Maas et al., 2011), ISEAR (Scherer and Wallbott, 1994), AMAN (Aman and Szpakowicz, 2008), TREC-6 (Li and Roth, 2002), and AGNews (Zhang et al., 2015). The templates are provided by Wang et al. (2023). The initial class-related word pools are drawn from Hu et al. (2022) and Zhu et al. (2024). Detailed information on the datasets, templates, and pool is given in Appx.D.

Experiment Settings We conduct few-shot learning experiments using LLaMA2-7b (Touvron et al., 2023) and GPT2-xl (Radford et al., 2019) to test the effectiveness of our methods.

For the baseline models, including **vanilla LLaMA2-7b**, **vanilla GPT2-xl**, **Topk** (Liu et al., 2022), **SelfICL** (Wu et al., 2023), and **DataICL** (Chang and Jia, 2023), we adopt their proposed demonstration samples and pair them with class name and our multiple class-related words, respectively, to create sample-label pairs. Detailed descriptions of the baseline models, including comparing their methods with ours and an overview of their experimental settings, are in Appx.E.

For each model, the sequential class-related word forward insertion is evaluated on the validation set, which is the full training set excluding the samples selected for demonstrations. In Vanilla, TopK, and SelfICL, the full training set is split into a reduced training set and a validation set in an 8:2 ratio as demonstration samples vary with test samples. The demonstration samples are selected based on the reduced training set, while the multiple class-related word insertion is based on the validation set. The ICL performance is evaluated on the same test set for all experiments.

4.2 Main Results

We present the results of each model and LICL (ours) under two settings: with and without multiple class-related word insertion in 1-shot ICL. The outcomes, summarized in Table 1, lead to the following conclusions:

LICL Outperforms Baselines: The proposed LICL consistently outperforms baseline methods across all datasets under the initial single-label pair conditions. It achieves an average accuracy improvement of 3.11% in LLaMA2-7b and 11.58% in GPT2-xl. The gains are especially significant in multi-class classification tasks, with notable improvements of 6.02% (AMAN, LLaMA2-7b), 12.80% (TREC, GPT2-xl), 12.53% (AMAN, GPT2-xl), and 23.68% (AGNews, GPT2-xl).

Multiple Class-Related Words Enhanced: In-

	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews	Avg.
LLaMA2-7b								
vanilla-LLaMA2-7b	93.06	93.24	94.81	68.20	53.08	70.91	81.78	79.30
+MLabels_CN	93.74↑	94.41↑	95.89↑	69.88↑	59.75↑	71.48↑	83.36↑	81.22
+MLabels_LW	93.74	94.41	95.89	72.88↑	59.05	71.03	83.88↑	81.55
TopK	92.37	92.82	94.29	77.80	51.38	64.32	79.09	78.87
+MLabels_CN	93.52↑	93.55↑	94.69↑	83.20↑	58.15↑	67.38↑	79.59↑	81.44
+MLabels_LW	93.52	93.55	94.69	83.20	58.65↑	66.98	79.59	81.45
SelfICL	91.71	93.35	94.69	76.00	53.63	65.18	81.46	79.36
+MLabels_CN	92.53↑	92.82↑	95.10↑	79.00↑	58.65↑	69.10↑	82.01↑	81.39
+MLabels_LW	92.53	92.82	95.10	**82.80**↑	58.65	69.30↑	82.01	81.96
DataICL	94.51	89.89	94.60	71.80	53.88	70.17	83.45	79.76
+MLabels_CN	95.28↑	92.55↑	94.70↑	74.40↑	55.64↑	71.30↑	84.83↑	81.40
+MLabels_LW	95.28	92.55	94.70	78.80↑	54.89	71.10	85.45↑	81.95
LICL	95.39[1]	94.41[1]	95.40[1]	78.40[1]	59.90[2]	72.49[2]	84.11[2]	82.87
+MLabels_CN	95.97[1]↑	95.15[1]↑	95.60[1]↑	79.80[1]↑	65.16[2]↑	**73.55**[2]↑	86.55[2]↑	84.54
+MLabels_LW	**95.97**[1]	**95.15**[1]	**95.60**[1]	80.60[1]↑	**69.40**[2]↑	73.09[2]	**86.58**[2]↑	**85.20**
GPT2-xl								
vanilla-GPT2-xl	71.74	64.26	67.00	46.84	29.97	39.00	55.24	53.44
+MLabels_CN	85.63↑	67.07↑	68.23↑	53.60↑	39.35↑	50.01↑	58.44↑	60.33
+MLabels_LW	85.63	67.07	70.13↑	54.76↑	39.30	50.56↑	58.24	60.81
TopK	69.41	65.69	63.36	56.20	32.83	44.12	54.33	55.13
+MLabels_CN	84.51↑	66.22↑	65.47↑	60.80↑	40.85↑	54.75↑	55.24↑	61.12
+MLabels_LW	84.51	66.22	67.07↑	61.80↑	40.60	54.88↑	55.24	61.47
SelfICL	70.07	64.89	60.96	56.00	32.58	44.98	54.50	54.85
+MLabels_CN	83.80↑	65.43↑	61.46↑	64.20↑	43.36↑	54.82↑	56.42↑	61.36
+MLabels_LW	83.80	65.43	62.16↑	65.40↑	43.11	55.08↑	56.42	61.63
DataICL	83.47	63.83	64.80	57.20	35.34	35.28	43.36	54.75
+MLabels_CN	84.84↑	65.74↑	69.80↑	58.20↑	36.84↑	48.04↑	51.14↑	59.23
+MLabels_LW	84.84	65.74	69.20	65.20↑	36.84	48.04	51.14	60.14
LICL	85.17[2]	64.89[1]	71.50[1]	70.00[2]	47.87[2]	48.64[2]	78.92[2]	66.71
+MLabels_CN	91.65[2]↑	**69.95**[1]↑	73.40[1]↑	70.40[2]↑	49.62[2]↑	**59.73**[2]↑	79.08[2]↑	**70.55**
+MLabels_LW	**91.65**[2]	**69.95**[1]	**73.40**[1]	**70.40**[2]	**49.87**[2]↑	58.74[2]	**79.49**[2]↑	70.50

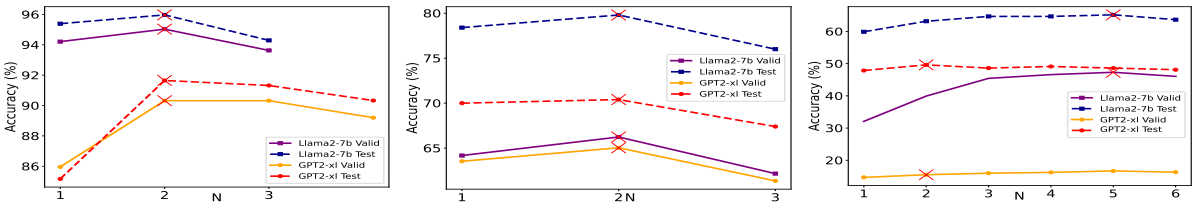
Table 1: 1-Shot ICL Experimental Results: Model Name (e.g., TopK, LICL) is demonstrated on initial sample-label pairs while '+MLabels' is on sample-multiple-label pairs. '+MLabels_CN' is predicting based on the maximum logit over class names, and '+MLabels_LW' is based on the maximum logit over inserted class-related words. The best accuracy results (%) are marked in bold. Marker [1] indicates results given under Eq.1, while marker [2] indicates results given under Eq.2. "↑" in '+MLabels_CN' signifies an increase in performance compared to the original method, while in '+MLabels_LW', it signifies an increase in performance compared to '+MLabels_CN'.

serting multiple class-related words ('+MLabels') significantly improves performance across all models. This strategy yields an average baselines' accuracy increase of 2.39% in LLaMA2-7b, reaching up to 2.6% in SelfICL, and an average increase of 5.94% in GPT2-xl, with a maximum improvement of 7.37% in vanilla GPT2-xl. Compared to configurations that employ a single class-related word per sample-label pair, LICL achieves notable accuracy gains, with increases of 2.33% in LLaMA2-7b (peaking at 9.6% in AMAN) and 3.84% in GPT2-xl (peaking at 11.09% in ISEAR). These improvements demonstrate significant enhancements over the initial single-label setup, underscoring the efficacy of the multiple class-related word approach.

Validation-Testing Results Alignment: Fig.3 displays the validation and test performance across various counts of inserted class-related words (N) for each dataset in LICL. Generally, the trends in test performance are consistent with those in validation performance. The peak test accuracy was achieved by incorporating multiple class-related words before reaching the stopping criterion, at which point the validation accuracy begins to decrease with further insertion of class-related words. This consistency supports the reliability of our stopping criterion in sequential forward insertion.

Initial Class-Related Word Updates Improved To boost the quality of the initial sample-label pairs, we update certain labels in two datasets based on the samples selected in LICL and DataICL before ICL experiments[3]. In AMAN, 'other' is replaced with 'neutral'. In TREC, 'entity' is replaced with 'animal', 'description' with 'definition', 'human' with 'persons', 'location' with 'state', and 'number' with 'numeric'. The enhanced accuracy pre-

[3]Updating initial class-related words ensures they have a high logit value relative to other words with the same label polarity to provide effective class-related guidance in demonstrations for all testing samples. However, Vanilla, TopK, and SelfICL use varying demonstration samples across test instances, leading to inconsistent evaluations and making updates impractical. Therefore, we use class names as initial words in the multiple insertion experiment.



(a) SST-2 (b) TREC (c) AMAN

Figure 3: Validation and test performance under inserted word quantity (N) in sample-multiple-label pairs. The red cross marks the reported result setting. In LLaMA2-7b, N is 2 for SST2, IMDB, TREC, ISEAR, and AGNews, 4 for CR, and 5 for AMAN. In GPT2-xl, N is 7 for ISEAR and 2 for others. The remaining datasets are in Appx.F.

sented in Table 2 highlights the effectiveness of class-related word updates over the direct use of predefined class names.

	LLaMA2-7b				*GPT2-xl*			
	DataICL		LICL		DataICL		LICL	
	original	update	original	update	original	update	original	update
AMAN	40.85	**53.88**	57.00	**59.90**	31.08	**35.34**	41.35	**47.87**
TREC	70.20	**71.80**	70.40	**78.40**	37.40	**57.20**	54.20	**70.00**

Table 2: Enhancement in 1-shot ICL accuracy (%) through label evaluation and updates.

4.3 Effectiveness of LICL in 5-shot ICL

To explore the capability of multiple class-related words further, we assess their effectiveness in a 5-shot setting using our method and SelfICL for one binary classification task (SST2) and one multi-class classification task (AMAN). For both methods, the 5-shot sample-label pairs are selected based on the top-5 scoring samples in the training set for each label. The validation set is created by removing all selected samples from the training set. The sample-label pairs are ordered based on the highest score achieved by each label among the selected samples.

	SST2	TREC	AMAN	Avg.	SST2	TREC	AMAN	Avg.
	Llama2-7b				*GPT2-xl*			
SelfICL	93.63	73.20	50.13	72.32	73.04	59.00	34.56	55.53
+MLabels_CN	94.89	74.00	55.89	74.93	86.60	69.20	40.40	65.38
+MLabels_LW	94.89	74.00	56.14	75.01	86.60	69.20	40.60	65.47
LICL	94.56	76.20	56.14	75.35	81.16	72.00	34.84	62.67
+MLabels_CN	95.39	78.60	58.40	76.90	89.18	**74.40**	51.80	**71.79**
+MLabels_LW	**95.39**	**78.60**	**58.40**	**76.90**	**89.18**	73.00	**51.80**	71.33

Table 3: Accuracy (%) with multiple class-related word insertion in 5-shot ICL: The scoring method and test sets are consistent with those in Table 1.

As shown in Table 3, the insertion of multiple class-related words is effective in 5-shot settings, yielding an average accuracy improvement of 2.69% and 9.93% in SelfICL, and 1.08% and 9.13% in LICL under LLaMA2-7b and GPT2-xl. LICL outperforms SelfICL across all results. Inserted words for both baseline and our models are in Appx.F, and enhancements for updating initial class-related words under 5-shot settings are in Appx.G.1.

4.4 Ablation Studies

4.4.1 Effectiveness of LICL Ordering

In LICL, we use the logit separability scores of samples as a decision feature to order sample-label pairs. To assess the effectiveness of this ordering strategy, we compare 1-shot ICL classification performance using LICL's initial sample-label pairs against 30 (LLaMA2-7b) and 50 (GPT2-xl) random permutations (excluding LICL's order) in multi-class tasks[4], and a flipped order in binary tasks.

	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews	Avg.
LLaMA2-7b								
LICL	**95.39**	**94.41**	**95.40**	**78.40**	59.90	72.49	84.11	**82.87**
Permutation	92.97	93.35	94.40	78.20	**60.15**	72.49	**85.82**	82.48
GPT2-xl								
LICL	**85.17**	**64.89**	71.50	**70.00**	**47.87**	48.64	**78.92**	**66.71**
Permutation	53.87	63.83	**82.20**	63.20	39.10	**52.03**	73.16	61.06

Table 4: Accuracy (%) comparison of our ordering method with random orders in multi-class datasets and the flipped order in binary datasets. 'Permutation' presents the best result for each dataset.

As shown in Table 4, LICL often outperforms or matches the best results among compared permutations. It achieves an average accuracy improvement of 0.39% in LLaMA2-7b and 5.65% in GPT2-xl. Although GPT2-xl shows a performance drop on the IMDB dataset, the binary nature of this task can mitigate this by allowing all possible orders to be tested exhaustively. LICL excels in multi-class tasks, matching or exceeding top permutation results. This underscores LICL's ordering strategy's efficiency, especially in multi-class tasks with thousands of possible orderings (7!). While enumerating all these orderings could take months, LICL

[4]We randomize the order of samples selected by LICL to create various permutations and then randomly select 30 or 50 different permutations from the generated pool. Each permutation is assessed individually in a 1-shot ICL classification.

achieves comparable performance within hours or minutes, demonstrating its practical advantage.

4.4.2 Effectiveness of Label Organization

In Table 1, inserting multiple class-related words significantly improves ICL performance across all models. To further evaluate the effectiveness of our label organization method, we compare it with a random selection and ordering of multiple class-related words. In Table 5, the lower accuracy observed in the '+MLabels-Random' highlights the advantage of our systematic approach. These results emphasize that thoughtful organization of class-related words is crucial for boosting ICL performance.

LICL LLaMA2-7b	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews
+MLabels-Random	94.84	93.62	94.90	79.40	57.64	70.56	80.49
+MLabels_LW	**95.97**	**95.15**	**95.60**	**80.60**	**69.40**	**73.09**	**86.58**

Table 5: Accuracy (%) comparison of our multiple class-related words organization method with random multiple words organization in P_r (same N as in Table 1).

4.5 Scalability to Larger LLMs

Table 6 presents the accuracy results of LICL implemented on LLaMA3-8b and LLaMA2-13b, validating the effectiveness and applicability of our method on larger language models. The insertion of multiple class-related words improved accuracy, with LICL achieving a 3.42% increase on LLaMA3-8b and 2.32% on LLaMA2-13b. Notably, for the IMDB dataset on LLaMA2-13b, despite significant sample information loss compared to other methods, our approach maintained strong performance. Details of the evaluation of initial class-related word updates can be found in Appx.G.2.

	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews	Avgs.
LLaMA3-8b								
LICL	95.66	94.29	95.50	79.00	61.15	70.83	84.67	83.01
+MLabels_CN	95.77	95.82	95.90	80.40	65.89	73.89	87.72	85.06
+MLabels_LW	**95.77**	**95.82**	**95.90**	**86.00**	**69.57**	**74.29**	**87.72**	**86.44**
LLaMA2-13b								
LICL	96.38	94.62	95.70*	82.20	60.15	73.16	85.12	83.90
+MLabels_CN	96.87	95.88	95.90*	84.00	67.64	74.37	87.79	86.06
+MLabels_LW	**96.87**	**95.88**	**95.90***	**84.40**	**67.89**	**74.82**	**87.79**	**86.22**

Table 6: Accuracy performance (%) of LICL on LLaMA2-13b and LLaMA3-8b. *: For the IMDB dataset on LLaMA2-13b, our GPU could only accommodate 20% of each demonstration sample length.

4.6 General Applicability of Inserted Words

To address potential concerns about overfitting when selecting multiple class-related words from a single dataset, we implement cross-dataset evaluations. We evaluate the efficacy of using two class-related words (N in most binary tasks in our study) selected from the SST2 dataset across the CR and IMDB datasets, as detailed in Table 7.

	CR	IMDB	CR	IMDB
	LLaMA2-7b		*GPT2-xl*	
LICL	94.41	95.40	64.89	71.50
+MLabels (from same dataset)	95.15	95.60	69.95	73.40
+MLabels (from SST2)	95.15	95.40	65.04	72.86

Table 7: Cross-dataset accuracy (%) evaluation on inserted multiple class-related words.

In LLaMA2-7b, performance improvements on the CR may be attributed to similar sub-token/token found in both SST2 and CR (e.g., " **un**healthy", " **un**fair", " **good**"). No performance decrease is observed in IMDB. In GPT2-xl, despite initial differences in the positive sentiment words across datasets, the class-related words from SST2 still enhanced performance in both CR and IMDB. These results indicate that the selected class-related words exhibit a level of generality, enhancing ICL performance across various datasets with similar labels.

4.7 Further Analyses

More in-depth analyses are in Appx.H, including pool filtering effectiveness and its ablation study (H.1), label bias evaluation (H.2), the impact of class-related words on predictions whether or not they appear in demonstrations (H.3), the effects of label-balanced demonstration in LICL (H.4), sample-label logit separability visualizations (H.5), and inference cost (H.6).

5 Conclusion

In this paper, we introduce logit separability, a criterion for measuring the instruction clarity of samples and labels based on their logit value distributions, assessing whether a sample is easily predictable by its true label and whether a label performs well across multiple samples of the same class. To improve ICL with clearer class-related guidance, we propose LICL, a logit separability-based method for selecting and ordering demonstrated samples and labels. Besides, to enrich ICL with broader label information, we present a novel label demonstration method by forming sample-label pairs through the insertion of multiple class-related words, guided by sequential forward search on the logits from selected samples and validation performance. Experimental results confirm that this combined approach significantly enhances ICL classification performance, yielding superior outcomes.

6 Limitations

In this paper, we enhance in-context learning performance by incorporating additional class-related words. Although related class-related words for various tasks have been extracted and collected, new datasets may still lack appropriate class-related words. However, powerful search tools such as WordNet (Pedersen et al., 2004), ConceptNet (Speer et al., 2017), and open-source vocabularies can mitigate this issue. Our method designs a filtering approach that refines the quality of class-related words based on these search results.

References

Saima Aman and Stan Szpakowicz. 2008. Using roget's thesaurus for fine-grained emotion recognition. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*.

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

Ting-Yun Chang and Robin Jia. 2023. Data curation alone can stabilize in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8123–8144.

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.

Yu Fei, Yifan Hou, Zeming Chen, and Antoine Bosselut. 2023. Mitigating label biases for in-context learning. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Keinosuke Fukunaga. 2013. *Introduction to statistical pattern recognition*. Elsevier.

Lingyu Gao, Debanjan Ghosh, and Kevin Gimpel. 2023. The benefits of label-description training for zero-shot text classification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13823–13844.

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

Shengding Hu, Ning Ding, Huadong Wang, Zhiyuan Liu, Jingang Wang, Juanzi Li, Wei Wu, and Maosong Sun. 2022. Knowledgeable prompt-tuning: Incorporating knowledge into prompt verbalizer for text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2225–2240.

Diana Kornbrot. 2014. Point biserial correlation. *Wiley StatsRef: Statistics Reference Online*.

Itay Levy, Ben Bogin, and Jonathan Berant. 2023. Diverse demonstrations improve in-context compositional generalization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1401–1422, Toronto, Canada. Association for Computational Linguistics.

Xiaonan Li and Xipeng Qiu. 2023. Finding support examples for in-context learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 6219–6235.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

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.

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, Dublin, Ireland. Association for Computational Linguistics.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Aristides Milios, Siva Reddy, and Dzmitry Bahdanau. 2023. In-context learning for text classification with many labels. In *Proceedings of the 1st GenBench Workshop on (Benchmarking) Generalisation in NLP*, pages 173–184.

Ted Pedersen, Siddharth Patwardhan, Jason Michelizzi, et al. 2004. Wordnet:: Similarity-measuring the relatedness of concepts. In *AAAI*, volume 4, pages 25–29.

Youri Peskine, Damir Korenčić, Ivan Grubisic, Paolo Papotti, Raphael Troncy, and Paolo Rosso. 2023. Definitions matter: Guiding gpt for multi-label classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4054–4063.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

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.

Klaus R Scherer and Harald G Wallbott. 1994. Evidence for universality and cultural variation of differential emotion response patterning. *Journal of personality and social psychology*, 66(2):310.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269.

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.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

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.

Xinyi Wang, Wanrong Zhu, Michael Saxon, Mark Steyvers, and William Yang Wang. 2024. Large language models are latent variable models: Explaining and finding good demonstrations for in-context learning. *Advances in Neural Information Processing Systems*, 36.

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*.

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 *2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022*, pages 2422–2437. Association for Computational Linguistics (ACL).

Zeping Yu and Sophia Ananiadou. 2024. How do large language models learn in-context? query and key matrices of in-context heads are two towers for metric learning. *arXiv preprint arXiv:2402.02872*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Yiming Zhang, Shi Feng, and Chenhao Tan. 2022. Active example selection for in-context learning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9134–9148.

Zixiao Zhu, Junlang Qian, Zijian Feng, Hanzhang Zhou, and Kezhi Mao. 2024. EDEntail: An entailment-based few-shot text classification with extensional definition. In *2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

A Exploring Logit Separability and Class-Related Word Effectiveness

A.1 Logit Separability Over Class-Related Words in Zero-Shot Learning

This study evaluates the logit value separability for negative and positive samples of class-related words 'bad' and 'pessimistic' compared to the class name 'negative', and for class-related words 'good' and 'happy' compared to the class name 'positive' in SST-2 under zero-shot learning. Except for the word bad', shown in Fig.1(b), the logit distribution figures for the remaining words are listed in Fig 4.

In Fig.1(b) and Fig.4, the first 100 samples are negative, while samples 101-200 are positive samples. Compared to the class names, the logit of negative class-related words across negative samples is higher than those for 'negative', while the logit for the same class-related words across positive samples is lower than 'negative'. Similarly, the logit for positive class-related words is higher across positive samples and lower across negative samples than 'positive', indicating better logit separability for these class-related words compared to their respective class names. This demonstrates superior logit separability for certain class-related words compared to class names. Since logit values are crucial for class prediction, this enhanced

Label Set	SST2	CR	TREC	AMAN	ISEAR
1	0: ‘0’, 1: ‘1’	0: ‘0’, 1: ‘1’	0: ‘0’, 1: ‘1’, 2: ‘2’, 3: ‘3’, 4: ‘4’, 5: ‘5’	0: ‘0’, 1: ‘1’, 2: ‘2’, 3: ‘3’, 4: ‘4’, 5: ‘5’, 6: ‘6’	0: ‘0’, 1: ‘1’, 2: ‘2’, 3: ‘3’, 4: ‘4’, 5: ‘5’, 6: ‘6’
2	0: ‘ negative’, 1: ‘ positive’	0: ‘ negative’, 1: ‘ positive’	0: ‘ abbreviation’, 1: ‘ entity’, 2: ‘ description’, 3: ‘ human’, 4: ‘ location’,5: ‘ number’	0: ‘ fear’, 1: ‘ sadness’, 2: ‘ disgust’, 3: ‘ anger’, 4: ‘ joy’, 5: ‘ surprise’, 6: ‘ others’	0: ‘ fear’, 1: ‘ sadness’, 2: ‘ disgust’, 3: ‘ anger’, 4: ‘ joy’, 5: ‘ guilt’, 6: ‘ shame’
3	0: ‘ bad’, 1: ‘ good’	0: ‘ bad’, 1: ‘ good’	0: ‘ abbreviation’, 1: ‘ animal’, 2: ‘ definition’, 3: ‘ persons’, 4: ‘ state’,5: ‘ numeric’	0: ‘ worry’, 1: ‘ sadness’, 2: ‘ loathing’, 3: ‘ rage’, 4: ‘ happy’, 5: ‘ stunning’, 6: ‘ neutral’	0: ‘ worry’, 1: ‘ grief’, 2: ‘ loathing’, 3: ‘ rage’, 4: ‘ happy’, 5: ‘ remorse’, 6: ‘ embarrassment’
4	0: ‘ terrible’, 1: ‘ great’	0: ‘ terrible’, 1: ‘ great’	0: ‘ abbreviation’, 1: ‘ food’, 2: ‘ reason’, 3: ‘ persons’, 4: ‘ city’, 5: ‘ count’	0: ‘ anxiety’, 1: ‘ sad’, 2: ‘ disgusting’, 3: ‘ angry’, 4: ‘ pleasure’, 5: ‘ surprising’, 6: ‘ noemo’	0: ‘ anxiety’, 1: ‘ sad’, 2: ‘ disgusting’, 3: ‘ angry’, 4: ‘ pleasure’, 5: ‘ regret’, 6: ‘ humiliation’

Table 8: The label information of each dataset.

separability can significantly improve classification performance.

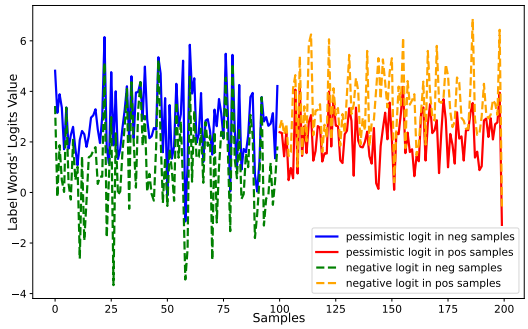
A.2 Label Effectiveness in ICL

A.2.1 Single Class-Related Word as Demonstration Label

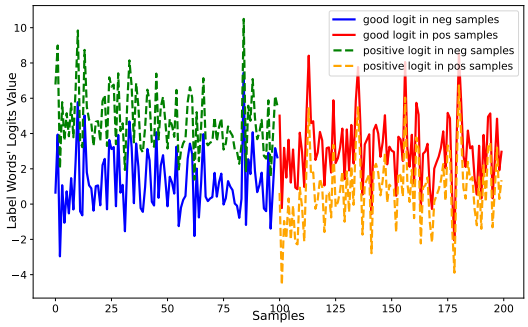
This study establishes four distinct label sets for each dataset, utilizing identical samples to form the sample-label pairs in 1-shot ICL, as detailed in Table 8. The reported results represent the average accuracy obtained from five repeated experiments, conducted with seeds 42, 43, 44, 45, and 46 in sample selection during the 1-shot demonstrations. As shown in Fig.5, in LLaMA2-7b, the maximum 49.20% accuracy difference (TREC), and the highest 11.08% (SST2) standard deviation are observed. We also evaluate the label effectiveness in GPT2-xl under the same experimental conditions, with results shown in Fig.6. The results are similar to those in LLaMA2-7b, with a maximum accuracy difference of 48.20% (TREC) and a maximum standard deviation of 10.52% (SST2). These findings suggest that certain class-related words with stronger logit separability, as exampled in Figs.1(b) and 4, can serve as more effective demonstration labels, contributing to greater accuracy and robustness in ICL than predefined class names.

A.2.2 Multiple Class-Related Words as Demonstration Label

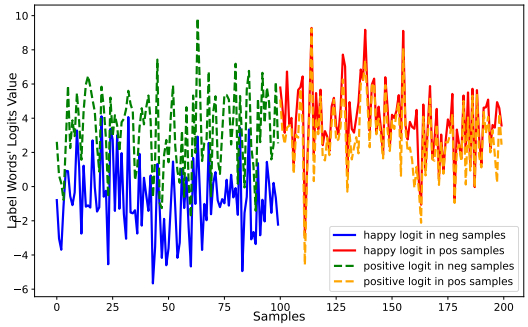
This study evaluates the performance of demonstrations using different numbers of class-related words. The multiple class-related words combine the class name with related class-related words, connected by spaces, such as "negative bad" and "positive good" in SST2 and CR. We also assess the label effectiveness in GPT2-xl under the same experimental conditions, with the results shown in Fig.7. These findings indicate the potential of using multiple class-related words in demonstrations to enhance ICL.



(a) ‘pessimistic’ vs ‘negative’



(b) ‘good’ vs ‘positive’



(c) ‘happy’ vs ‘positive’

Figure 4: Class-related words logit separability over samples.

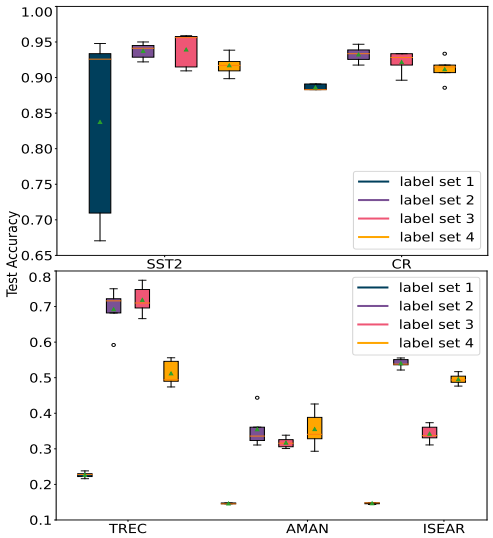


Figure 5: Label effectiveness in ICL (GPT2-xl)

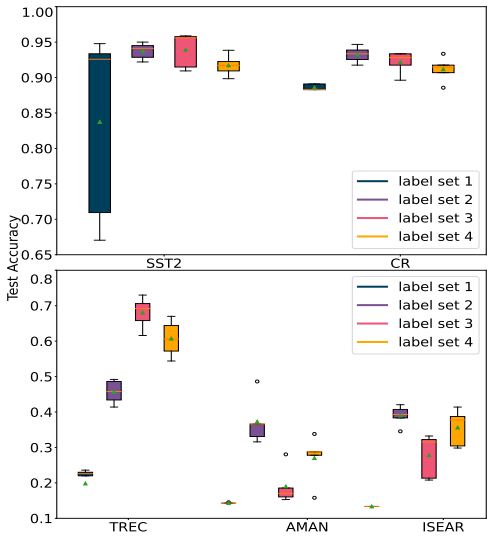


Figure 6: Label effectiveness in ICL (GPT2-xl)

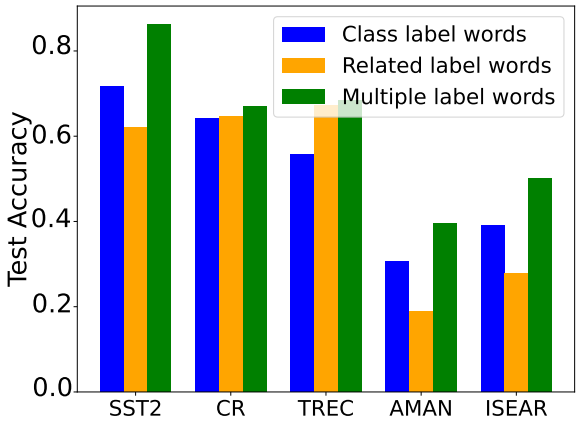


Figure 7: Multiple class-related words effectiveness in ICL (GPT2-xl)

B Class-Related Word Pool Filtering

The goal of class-related word pool filtering is to enhance the logit separability of the words within the pool, guided by logit feedback from LLM across the dataset. Ideally, each word should exhibit a clearly dominant logit value in samples with the corresponding label, compared to those with different labels, to prevent the introduction of incorrect class-related features in the ICL demonstration.

To achieve this, we first evaluate the separability of each class-related word based on its logit value across the training set D in a zero-shot setting, where input samples are formatted using a template without labels. For a word $w \in P$ with label $l \in L$, we exclude it if its average logit b_l for training samples labeled with l is not the highest among all training samples. This step ensures that the class-related word provides accurate logit feedback in the corresponding training samples compared to those with different label expressions. To further ensure the separability of the feedback, avoiding situations where, for example, a class-related word has an average logit of 7.1 in correct training samples but a close value of 7.0 in semantically unrelated samples, we apply point-biserial testing to test the correlation r between the word's logit values in samples with the same label versus those with different labels as defined in Eq.4:

$$r = \frac{V_0 - V_1}{\sigma} \times \sqrt{\frac{n_0 \times n_1}{(n_0 + n_1)^2}} \quad (4)$$

where V_0 denotes the mean logit values of the class-related word for training samples with the same label, V_1 denotes the mean logit values for samples with different labels, σ represents the standard deviation of the logit values, and n_0 and n_1 correspond to the number of samples with the same and different labels, respectively.

For a given class-related word, a positive correlation $r > 0$ indicates that its average feedback across training samples within its own class significantly differs from that of samples in other classes, warranting its retention. Otherwise, the word will be removed from the pool. This process ensures that the remaining words in the refined pool P_r are not only logit separable but also exhibit clear class-related information distinctions.

C Prompt Examples

The prompts used in this paper are exampled below:

Zero-Shot Learning:

```
Review: they 're easy to use
Sentiment:
```

One-Shot Learning:

```
Review: norton support is completely
pathetic
Sentiment: negative
Review: overall , i am very pleased
with it
Sentiment: positive
Review: they 're easy to use
Sentiment:
```

One-Shot Multiple Class-Related Word Insertion Learning (result from CR in LlaMA2-7b):

```
Review: it does not only have
difficulty playing jpegs , it even
has trouble ...
Sentiment: negative unhealthy unjust
Review: about the product the zen
micro is a sleek , stylish ...
Sentiment: positive good favorable
Review: they 're easy to use
Sentiment:
```

D Statistic Information of Datasets and Templates

The statistic information of datasets and templates are listed in Table 9. Suppose the original dataset has no train/test split. In that case, a testing set is randomly selected, comprising 20% of the entire dataset with a balanced-label distribution, while the remaining data is used for training (AMAN, ISEAR). If the dataset includes a validation set, the original training and validation sets are combined to form a complete training set for demonstration selection (SST2). For AGNews, only 4,000 training samples are selected, with 1,000 samples per label, due to memory constraints.

The class-related word pools for SST2, CR, IMDB, and AGNews are derived from Hu et al. (2022), while those for TREC, AMAN, and ISEAR are sourced from Zhu et al. (2024).

E Experimental Implementation Details

All experiments are implemented under Python 3.8 environment and PyTorch 2.1.0. with Cuda version 11.8, GPU NVIDIA RTX A5000.

Baseline Model Experimental Settings The detailed information on the baseline models and the corresponding experimental settings for few-shot learning experiments is provided below. For all experiments conducted with LLaMA2-7b, LLaMA3-13b, and LLaMA3-8b, the model is configured to operate under a 4-bit setting.

Dataset	Template	Class Name	#Train	#Validation	#Test
SST2	Review: Sentiment:	positive, negative	6920	872	1821
CR	Review: Sentiment:	positive, negative	3394	-	377
IMDB	Review: Sentiment:	positive, negative	1000	-	1000
AMAN	Review: Emotion:	angry, disgust, joy, others, surprise, sad, and fear	4090	-	-
ISEAR	Review: Emotion:	angry, disgust, joy, shame, guilt, sadness, and fear	7666	-	-
TREC	Question: Answer Type:	location, number, description, entity, human, and abbreviation	5451	-	490
AGNews	Article: Answer:	Worlds, Business, Sports, and Technology	120000	-	7600

Table 9: The applied template and statistic information in each dataset.

Vanilla LLaMA2-7b (Touvron et al., 2023): We use LLaMA2-7b[5], a 7 billion parameter language model with 4096 tokens available. Prompts exceeding the model's token limit are truncated in the few-shot settings. The demonstrations are randomly selected and ordered on each label using five random seeds: 42, 43, 44, 45, and 46. The reported results are the average ICL accuracy over five runs.

Vanilla GPT2-xl (Radford et al., 2019): We use GPT2-xl[6], a 1.5 billion parameter language model with 1024 tokens available. Prompts exceeding the model's token limit are truncated. The demonstration and results settings are the same as Vanilla LLaMA2-7b.

TopK (Liu et al., 2022): An unsupervised method selects the nearest neighbors of the test samples as the demonstration samples using S-BERT[7]. In the re-run experiment, we choose samples for each label in the order ranked by their semantic similarity to the test sample.

SelfICL (Wu et al., 2023): A supervised method selects demonstration samples via S-BERT and ranks them based on Minimum Description Length (MDL). In the re-run experiment, after selecting the candidates, we randomly choose 30 combinations (the default setting) containing one sample for each label for MDL ranking with a window size 10. The best results are used as the selected-and-ranked demonstrations for ICL testing.

[5] https://huggingface.co/meta-llama/Llama-2-7b-chat-hf

[6] https://huggingface.co/openai-community/gpt2-xl

[7] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

DataICL (Chang and Jia, 2023): A supervised method trains a linear regressor to fit the LLM's output based on which sample is present and its order in the demonstration. In the re-run experiment, we select the sample with the highest score in each label as the demonstration samples, following the resulting order. The LLM used in DataICL is the same as the ICL evaluation model.

Table 10 indicates whether each baseline and our proposed method incorporate functions for the selection and ordering of samples and labels, demonstrating that our one-stop method encompasses all components of demonstration organization.

	Sample		Label	
	Selection	Ordering	Selection	Ordering
Vanilla	X	X	X	-
TopK	✓	✓	X	-
SelfICL	✓	✓	X	-
DataICL	✓	X	X	-
LICL(ours)	✓	✓	✓	✓

Table 10: Assessment of Demonstration Organization Methods in the Selection and Ordering of Samples and Labels. A '✓' indicates that the method incorporates this feature, an 'X' signifies its absence, and a '-' indicates that the feature is not applicable.

F The Number of Inserted Labels Settings

The number of class-related words (N) used in Table 1 on the baseline models are summarized in Table 11.

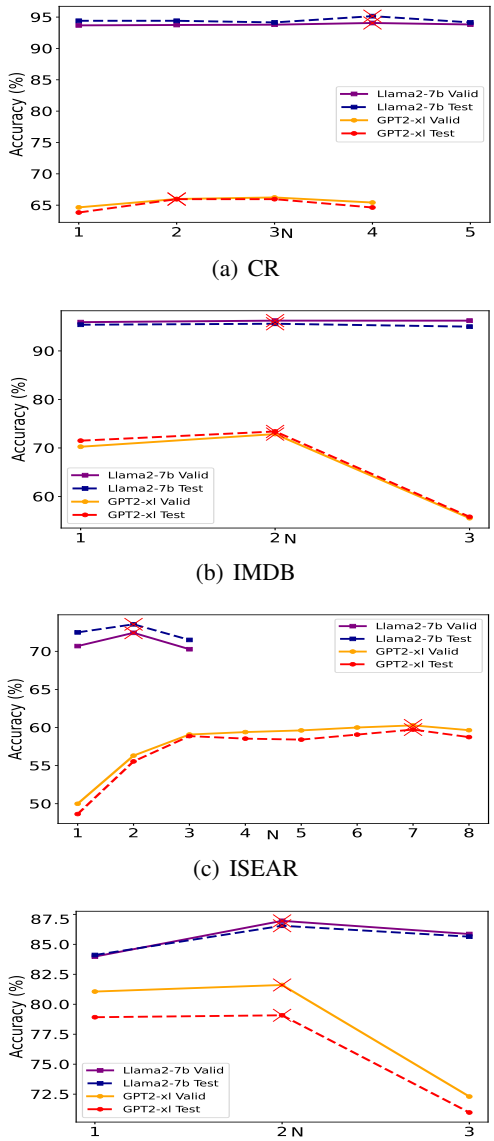
	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews
LLaMA2-7b							
vanilla-LLaMA2-7b	2	5	4	2	5	6	5
TopK	2	2	2	2	6	2	2
SelfICL	2	2	2	2	4	3	2
DataICL	2	2	2	2	3	6	6
LICL	2	4	2	2	5	2	2
GPT2-xl							
vanilla-GPT2-xl	3	2	2	4	3	6	2
TopK	3	3	2	4	6	5	2
SelfICL	3	2	2	3	6	5	2
DataICL	2	2	2	2	2	4	2
LICL	2	2	2	2	2	7	2

Table 11: The number of class-related words (N) inserted in the demonstration in the baseline models and LICL (ours) under each dataset in 1-shot ICL.

The remaining datasets' validation and test accuracy performance like Fig.3 is shown in Fig.8.

The number of class-related words (N) used in Table 3 on the baseline models are summarized in Table 12.

For simple classification tasks (binary tasks) such as SST2, CR, and IMDB, inserting around 2 to 3 class-related words yields good performance.



(a) CR

(b) IMDB

(c) ISEAR

(d) AGNews

Figure 8: Validation and test performance under different class-related word quantity (N) in sample-multiple-label pairs for CR. IMDB, ISEAR and AGNews.

In contrast, for fine-grained tasks (multi-class tasks) such as AMAN, ISEAR, and AGNews, inserting more class-related words is necessary to achieve better performance. Additionally, the larger language model (LLaMA2-7b) can effectively handle more label information compared to the smaller language model (GPT2-xl).

G Enhancement via Initial Class-Related Word Updates

G.1 5-Shot ICL

Table 13 presents the effectiveness of initial class-related word updates in 5-shot experiments on the

	SST2	AMAN	SST2	AMAN
	LLaMA2-7b		*GPT2-xl*	
SelfICL	2	6	4	4
LICL	2	3	3	5

Table 12: The number of class-related words (N) inserted in the demonstration in SelfICL and LICL (ours) under SST2 and AMAN in 5-shot ICL.

AMAN and TREC datasets, using the same word update information as in the 1-shot setting, as described in Sec.4.2.

LICL	AMAN	TREC	AMAN	TREC
	LLaMA2-7b		*GPT2-xl*	
Original	42.35	70.00	30.83	64.60
Update	56.14	76.20	34.84	72.00

Table 13: Enhancement in 5-shot ICL accuracy (%) through label evaluation and updates.

G.2 Larger LLMs

In the LLaMA3-8b and LLaMA2-13b experiments, we update certain labels in AMAN and TREC based on samples selected in LICL prior to the ICL experiments. In AMAN, 'other' is replaced with 'noemotion' in LLaMA3-8b and with 'neutral' in LLaMA2-13b. In TREC with LLaMA3-8b, 'entity' is replaced with 'element' 'description' with 'definition,' 'human' with 'individual,' and 'location' with 'place'. In LLaMA2-13b, 'entity' is replaced with 'word', and 'human' with 'persons'. Table 14 shows the improved accuracy across models: AMAN and TREC experience increases of 3.8% and 5.2% respectively in the LLaMA3-8b model, while AMAN and TRED see increases of 1.28% and 6.6% respectively in the LLaMA2-13b model. These results demonstrate the superiority of our initial class-related word replacement method over the direct use of class names in enhancing 1-shot ICL performance, even with larger language models.

LICL	AMAN	TREC	AMAN	TREC
	LLaMA3-8b		*LLaMA2-13b*	
Original	57.27	73.80	58.87	75.60
Update	**61.15**	**79.00**	**60.15**	**82.20**

Table 14: Enhancement in 1-shot ICL accuracy (%) through initial class-related words updates in LLaMA3-8b and LLaMA2-13b.

H Further Studies

In this section, we conducted more detailed experiments to analyze the effectiveness of our proposed approach for inserting multiple class-related words into demonstrations.

H.1 Case Study: Effectiveness of Pool Refinement

Table 15 lists the number of filtered words (-) for each dataset under two LLMs during point-biserial testing (P_r). Numerous words filtered indicate that although many words match the task topic definition at the linguistic level, they are not suitable as class-related words at the LLM level. This suggests that simple label-based voting, commonly used in many prompt-based methods, might harm LLM's ICL, as candidate words do not align with the task based on LLM's understanding.

Model	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews
LLaMA2-7b	-316	-237	-240	-32	-78	-78	-1163
GPT2-xl	-397	-375	-389	-42	-68	-118	-648

Table 15: The statistic information of filtering results under two-stage filtering.

The point-biserial filtering method refines class-related word pool, influencing the selection of demonstration samples. We analyzed its effect by comparing samples selected with and without it. Findings indicate notable impacts on sample selection for the CR (in both LLaMA2-7b and GPT2-xl) and AMAN (in GPT2-xl) datasets. Table 16 presents the comparative 1-shot ICL classification results, demonstrating that point-biserial filtering not only enhances sample selection but also boosts classification performance. Omitting this method reduces performance, confirming its effectiveness. Besides, even without this filtering, the addition of multiple class-related words still improves performance.

LICL	CR	CR	AMAN
	LLaMA2-7b	*GPT2-xl*	*GPT2-xl*
With rpb	94.41	64.89	47.87
+MLabels_CN	95.15	69.95	49.62
Without rpb	93.35	64.36	40.60
+MLabels_CN	94.68	65.43	48.12

Table 16: 1-shot ICL classification accuracy performance (%) comparing scenarios with ("With rpb") and without ("Without rpb") point-biserial filtering.

H.2 Case Study: Multiple Label Biases Evaluation

To address concerns regarding label biases with the insertion of multiple class-related words, we first evaluate the robustness by examining the standard deviation performance from baseline models using both single and multiple label insertions across five random sample iterations. Table 17 summarizes the results.

	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews
LLaMA2-7b							
Single Class-Related Word	2.81e-4	1.29e-4	8.77e-5	2.01e-3	9.97e-4	5.38e-4	9.69e-4
Multiple Class-Related Words	1.37e-4	8.70e-5	4.52e-5	1.31e-3	8.09e-4	2.40e-4	9.25e-5
GPT2-xl							
Single Class-Related Word	1.03e-2	1.63e-05	1.34e-2	2.23e-03	5.15e-3	8.17e-4	2.68e-3
Multiple Class-Related Words	1.47e-3	5.66e-06	9.03e-3	1.24e-03	4.84e-3	7.20e-4	1.55e-3

Table 17: Standard Deviation Evaluation: Results from experiments in vanilla and vanilla + Demo-MLabels_LW configurations.

Our findings suggest that using multiple class-related words in our demonstrations does not introduce new label biases, but rather enhances robustness, as indicated by a lower standard deviation compared to the single label approach. Additionally, as shown in Table 1, the classification performance has improved under both single and multiple class-related words scenarios compared to the baseline models.

We also conducted an additional experiment for comparison with Domain-context calibration (DC) (Fei et al., 2023), a label bias calibration method. In this experiment, the DC model is applied to GPT2-xl across three datasets: SST2, TREC, and AGNews. The training and testing conditions are the same as our approach to ensure a fair comparison. The in-context examples in DC are 1-shot and label-balanced, similar to our setup. Other settings, such as the number of seeds, follow the default settings[8].

Table 18 details the accuracy performance for DC and our method (LICL). Our method outperforms DC across the evaluated datasets, highlighting its effectiveness in improving classification accuracy in varied contexts without increasing label bias.

H.3 Case Study: Analysis of Utilizing Multiple Class-Related Words Mapping in Prediction

Table 1 and 3 show that '+MLabels_LW' (prediction based on the maximum logit over inserted class-related words) achieves higher accuracy than

[8] https://github.com/fywalter/label-bias

GPT2-xl	SST2	TREC	AGNews
DC	85.60	58.90	76.90
LICL	85.17	70.00	78.92
+MLabels_CN	91.65	70.40	79.08
+MLabels_LW	**91.65**	**70.40**	**79.49**

Table 18: Accuracy Performance (%) Comparison between DC and LICL on GPT2-xl.

'+MLabels_CN' (prediction based on the maximum logit over class names) in some datasets. This seems to align with the idea that incorporating extra class-related word mapping in the final prediction can enhance prompt-based methods (Schick and Schütze, 2021). However, in ICL, models tend to predict based on the labels provided in demonstrations, which may perform differently with extra class-related word mappings in prediction. We compare the effectiveness of using extra class-related word mappings only in predictions versus including them in demonstrations for both binary-class and multi-class tasks. Additionally, we evaluate the impact of leveraging extra class-related word mappings under zero-shot settings.

	SST2	TREC	AMAN	ISEAR	AGNews	Avg.
LLaMA2-7b						
ZSL	88.96	68.80	48.87	58.60	67.29	66.50
ZSL_P_r	90.94	60.00↓	47.37↓	60.80	60.03↓	63.83
LICL	95.39	78.40	59.90	72.49	78.06	76.85
LICL_P_r	95.44	72.00↓	63.13	71.56↓	74.37↓	75.30
+MLabels_CN	95.97	79.80	65.16	73.55	86.55	80.21
+MLabels_LW	95.97	80.60	69.40	73.09	86.58	81.13
+MLabels_P_r	94.51↓	74.40↓	67.64↓	71.56↓	75.33↓	76.69
GPT2-xl						
ZSL	79.57	38.00	39.60	42.33	53.16	50.53
ZSL_P_r	50.30↓	46.80	37.84↓	42.52	51.29↓	45.75
LICL_C	85.17	61.60	47.87	48.64	78.92	64.44
LICL_P_r	85.17	67.60	47.62↓	48.70	65.28↓	62.87
+MLabels_CN	91.65	70.40	49.62	59.73	79.08	70.10
+MLabels_LW	91.65	70.40	49.87	58.74	79.49	70.03
+MLabels_P_r	91.65	68.80↓	49.87	58.64↓	67.58↓	67.31

Table 19: Impact of prediction under class names, multiple class-related words in demonstration and P_r. Arrow ↓ indicates a decrease in accuracy of predictions over P_r, compared to those over class names (CN) or inserted class-related words (LW).

In Table 19, most experiments show decreased performance when predicting based on class-related words in P_r compared to predictions on class-related words that appeared in demonstrations (including class names). This suggests that simply applying extra class-related word mappings in the final prediction might disrupt information learned from demonstrations in ICL. Surprisingly, even under zero-shot learning (ZSL), where la-

bel information isn't prompted, adding extra label knowledge still decreases some classification performance. These results highlight the complex role of labels in ICL classification.

H.4 Case Study: Effectiveness of Label Balance in LICL

Our method is evaluated under a label-balanced demonstration setting, assuming that every category of label information matters. We also investigate an unbalanced setting by removing a sample-label pair with the highest sample score according to our scoring methods. Table 20 summarizes the results for two language models in 1-shot ICL.

	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews	Avg.
LLaMA2-7b								
LICL	95.39	94.41	95.40	78.40	59.90	72.49	84.11	82.87
LICL unbalanced	91.98	93.35	93.60	68.00	57.39	70.70	81.49	79.50
+MLabels_CN	95.97	95.15	95.60	79.80	65.16	**73.55**	86.55	84.54
+MLabels_CN unbalanced	92.42	93.62	94.40	69.60	62.16	71.36	85.87	81.35
+MLabels_LW	**95.97**	**95.15**	**95.60**	80.60	**69.40**	73.09	**86.58**	**85.20**
+MLabels_LW unbalanced	92.42	93.62	94.40	81.80	62.91	71.36	86.36	83.27
GPT2-xl								
LICL	85.17	64.89	71.50	70.00	47.87	48.64	78.92	66.71
LICL unbalanced	51.02	73.67	67.80	53.60	41.60	54.62	34.45	53.82
+MLabels_CN	91.65	69.95	73.40	70.40	49.62	**59.73**	79.08	**70.55**
+MLabels_CN unbalanced	52.85	74.47	68.20	55.40	45.86	56.75	43.75	56.75
+MLabels_LW	**91.65**	69.95	**73.40**	**70.40**	**49.87**	58.74	**79.49**	70.50
+MLabels_LW unbalanced	52.85	**74.47**	68.40	59.60	45.61	56.42	43.75	57.30

Table 20: Accuracy performance (%) of LICL, '+MLabels_CN', '+MLabels_LW' under label-balanced and label-unbalanced demonstrations.

The impact of label-unbalanced demonstrations varies between the two models. For LLaMA2-7b, a large-size language model, the negative effects of missing label information are less marked, with an average accuracy decrease of 3.37%. The most significant accuracy drops in TREC by 10.40%. The leverage of multiple class-related words mitigated the performance loss, reducing the average accuracy decline to 1.93% in '+MLabels_LW' compared to a label-balanced setting. In contrast, GPT2-xl, a smaller model with 1.5 billion parameters, experienced a marked performance decline under unbalanced conditions, averaging a 12.89% decrease in accuracy. Specifically, SST2 and AGNews experienced over 35% declines. Interestingly, CR demonstrated improved performance despite the unbalanced labels. The incorporation of multiple class-related words consistently enhanced performance across all datasets in the unbalanced setting, affirming their utility in ICL.

H.5 Case Study: Sample-Label Logit Separability Visualization

To assess the effectiveness of sample selection based on the discriminative logit separability of class-related words and the insertion of class-related words into test samples, Fig.9 illustrates the distribution of class-related word logit for selected and other samples, while Fig.10 displays the logit separability of inserted class-related word logit under 1-shot ICL learning across testing samples, along with other lass-related word logit separability under the same conditions.

Comparing Fig.9(a) with Fig.9(c), and Fig.9(b) with Fig.9(d), it is evident that the selected samples demonstrate superior logit separability between corresponding polarity class-related words and those of the opposite polarity, which enhances class prediction accuracy.

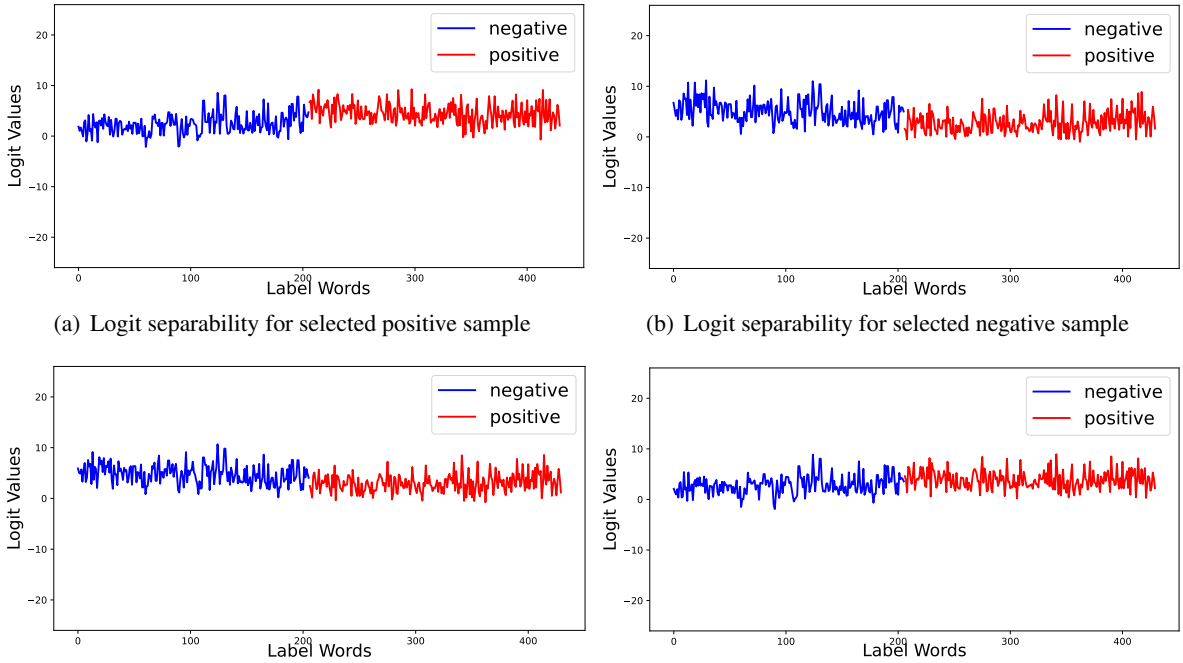
Similarly, an analysis of Fig.10(a) against Fig.10(c), and Fig.10(b) against Fig.10(d), shows that selected words exhibit improved logit separability between samples of matching polarity and those of opposing polarity, contributing positively to class prediction.

H.6 Case Study: Inference Cost Evaluation

To address concerns about the inference costs associated with the insertion of additional multiple class-related words, we evaluated the inference time of all compared methods and ours in 1-shot ICL experiments. The primary factors affecting inference costs are the length of the demonstration sample and the size of the language model. Since all demonstrations are 1-shot and balanced in terms of label distribution, variations in sample length directly impact inference times. Adding multiple class-related words increases the length of label inputs beyond using just the class name.

Table 21 details the running times for test sets using both our method and comparative models, and includes the incremental times associated with increasing the number of class-related words (in Figs. 3 and 11) in demonstrations, marked as '+MLabels time increased'.

Under 1-shot demonstration conditions, the inference times for our method are comparable to, or even faster than, those of other methods, given the length of the selected samples. With the addition of multiple labels, the inference time increases by minutes or seconds, which we consider to be within an acceptable range, given the enhanced



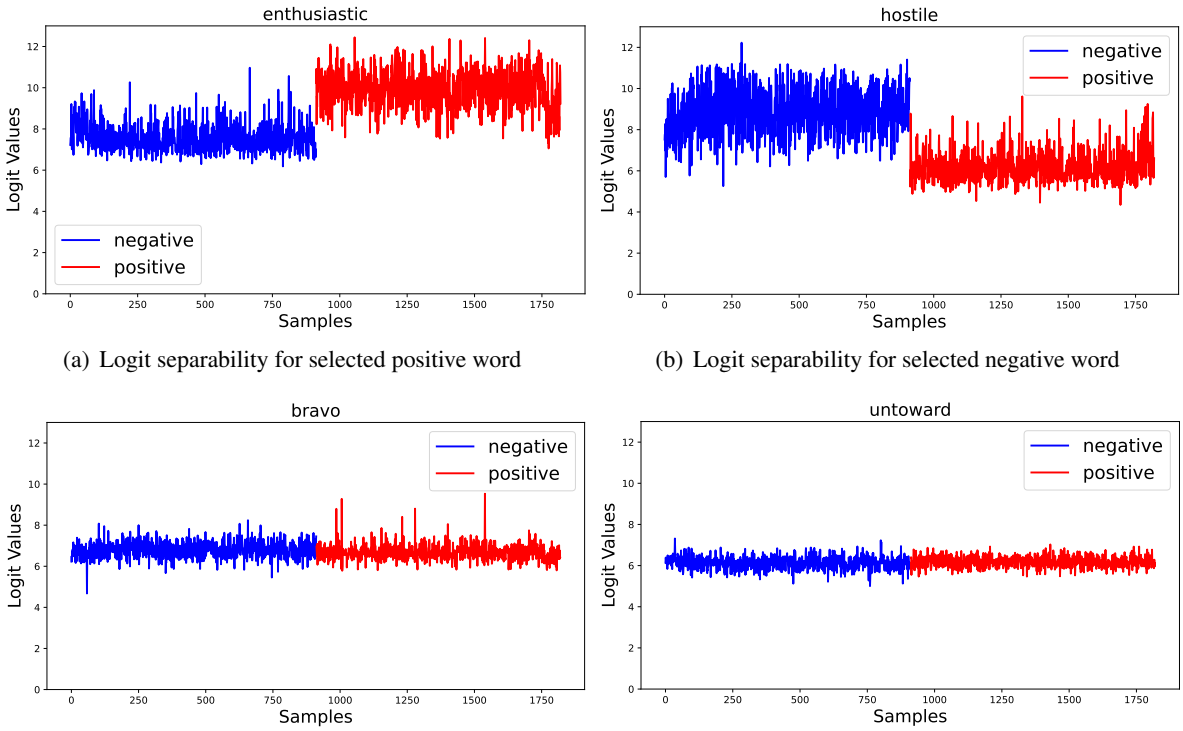
(a) Logit separability for selected positive sample

(b) Logit separability for selected negative sample

(c) Logit separability for a positive sample

(d) Logit separability for a negative sample

Figure 9: Visualization of samples' logit separability across class-related words in LLaMA3-8b, evaluated on the SST2 training set using a refined verbalizer. The class-related words are categorized into negative (0-205) and positive (206-429) groups.



(a) Logit separability for selected positive word

(b) Logit separability for selected negative word

(c) Logit separability for a positive word

(d) Logit separability for a negative word

Figure 10: Visualization of class-related words' logit separability in LLaMA3-8b, evaluated on the SST2 testing set using a refined verbalizer. The samples are categorized into negative (0-911) and positive (912-1820) groups.

Llama2-7b	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews	*GPT2-xl*	SST2	CR	IMDB	TREC	AMAN	ISEAR	AGNews
Topk	10m46s	2m35s	55m27s	2m39s	3m46s	21m48s	1h40m32s	Topk	1m53s	22s	6m8s	31s	50s	5m55s	24m30s
SelfICL	11m9s	2m19s	49m46s	3m	3m45s	19m11s	1h44m37s	SelfICL	2m	24s	6m18s	32s	48s	4m49s	24m55s
DataICL	7m5s	1m58s	52m32s	2m37s	3m4s	15m51s	1h34m37s	DataICL	1m56s	24s	6m13s	31s	43s	6m25s	18m46s
LICL	9m41s	2m12s	38m52s	2m31s	3m22s	15m30s	1h29m54s	LICL	1m26s	22s	6m20s	21s	41s	4m45s	19m7s
+MLabels time increased	+2s	+6s	+58s	+3s	+16s	+18s	+215s	+MLabels time increased	+1s	+1s	+7s	+2s	+1s	+10s	+26s

Table 21: Inference Time Evaluation (where 'h' denotes hours, 'm' denotes minutes, and 's' denotes seconds).

performance and accuracy achieved.