Incomplete In-Context Learning

Wenqiang Wang[1,τ], **Yujia Wen**[1,τ], **Yan Xiao**[1], **Zhifeng Chen**[1],
Yangshijie Zhang[3], Peng Chen[1], Mingbo Yang[1], Xiaochun Cao[1,2*]
[1]Shenzhen Campus of Sun Yat-sen University, [2]Peng Cheng Laboratory
[3]Lanzhou University
wangwq69@mail2.sysu.edu.cn,
caoxiaochun@mail.sysu.edu.cn

Abstract

Existing *In-context Learning (ICL)* typically assumes the retrieval dataset contains demonstrations for all output label spaces. However, in real-world scenarios, delays in dataset updates or incomplete data annotation may result in the retrieval dataset containing labeled demonstrations for only a subset of the output space. We refer to this phenomenon as an *incomplete retrieval dataset* and define the in-context learning under this condition as *Incomplete In-context Learning (IICL)*. To address IICL, we propose *Iterative Judgments and Integrated Prediction (IJIP)*, a framework with train-free and train-based variants. For classification, the iterative judgments stage of IJIP reformulates an m-class problem into m binary tasks, converting IICL into standard ICL. The integrated prediction stage of IJIP then refines results using both the input and initial predictions. We further extend IJIP to text regression and generation, and introduce lightweight variants that reduce computation and token costs. Across six LLMs, seven tasks, and eight datasets, IJIP achieves state-of-the-art results under two incompleteness settings and even outperforms standard ICL with complete labels. IJIP also supports a semi-supervised variant and can serve as a plug-and-play enhancement for existing ICL and zero-shot methods.

1 Introduction

Large language models (LLMs) achieve strong downstream performance through *in-context learning* (ICL) (Bertsch et al., 2025; Nafar et al., 2025; Ho et al., 2024), using demonstrations (Li et al., 2025e; Purohit et al., 2025) retrieved from the retrieval dataset (Pham et al., 2025; Peng et al., 2025). The effectiveness of ICL relies critically on the availability of labeled demonstrations covering all possible output classes (Kossen et al., 2024; Li et al., 2025a).

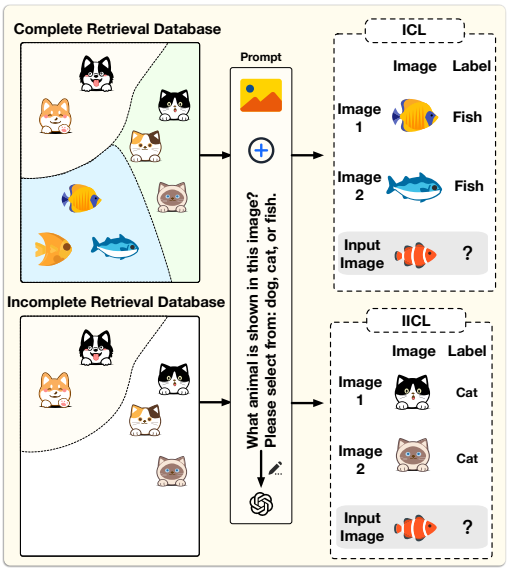


Figure 1: Comparison of complete vs incomplete retrieval dataset and ICL vs IICL scenarios. In the *incomplete retrieval database*, "fish" images are absent; thus, IICL fails to retrieve suitable demonstrations but retrieves "cat" images when the input is "fish", limiting IICL's performance.

Prior work commonly assumes retrieval datasets contain annotated demonstrations for all labels (**?**Momeni et al., 2025), an assumption frequently violated in practice. For example, new labels may emerge before database updates (Bell et al., 2025; Kumaravelu et al., 2025); extreme class imbalance (Haixiang et al., 2017) (Li et al., 2025c; Mildenberger et al., 2025) can leave certain labels without demonstrations; or Positive-Unlabeled (PU) learning (Li et al., 2009) (Kumagai et al., 2025; Mansouri et al., 2025) settings may lack full annotations. These issues yield a retrieval dataset covering only a subset of the full label space (Fig. 1), a condition we term the *incomplete retrieval dataset*. In-context learning under this setting is referred to as *Incomplete In-context Learning (IICL)*. Empirical results (Section 2.1)

*Corresponding author: Xiaochun Cao

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 41629–41650
July 2-7, 2026

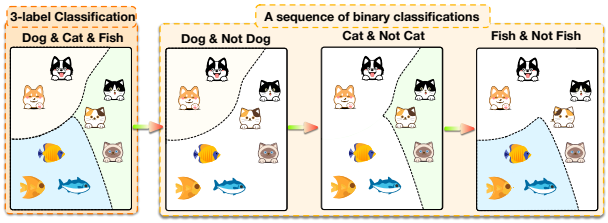


Figure 2: Comparison of m-label classification and a sequence of binary classifications. We transform a three-class classification task, "What's this? Choose from dog, cat, and fish." into a sequence of binary classifications consisting of the following sub-questions based on the input image: (1) "Is this a dog?" (2) "Is this a cat?" and (3) "Is this a fish?".

show IICL performance degrades as the proportion of missing labels increases, yet consistently exceeds zero-shot prompting. Notably, performance drops significantly even with one missing label, underscoring the importance of addressing incomplete retrieval scenarios in IICL.

To address IICL and the incomplete retrieval dataset, we propose *Iterative Judgments and Integrated Prediction (IJIP)* method, which includes both training-free and training-based variants (Liu et al., 2024a; Zhu et al., 2024). The training-free IJIP operates in two stages: *Iterative Judgments Stage* and *Integrated Prediction Stage*. In *Iterative Judgments Stage*, we first retrieve the top-k most semantically similar demonstrations from the retrieval dataset for the test text. As Tab. 1 shows, if a data is labeled as C_j, it is also implicitly not C_1, C_2, $\cdots$, C_{j-1}, C_{j+1}, $\cdots$, C_m. This can be formally expressed as: $\overline{C_1}$, $\overline{C_2}$, $\cdots$, $\overline{C_{j-1}}$, $\overline{C_{j+1}}$, $\cdots$, $\overline{C_m}$. This reformulates an m-class classification into m binary tasks (see Fig. 2), where the j-th task determines whether the input belongs to C_j or $\overline{C_j}$. To implement this, we modify the prompt in the LLMs as follows:

"Based on this image, answer m sub-questions. Is the label of this image C_1? Is the label of this image C_2? $\cdots$ Is the label of this image C_m? For example, input data:  *, labels: Dog, Not cat, Not fish; Input data:*  *, labels: Not dog, Cat, Not fish; Input data:*  *... "*

We assume the incomplete retrieval dataset contains annotated data for w labels, where $w < m$, meaning data for $m - w$ labels is missing in the original m-class setting. This issue is resolved in the binary classification formulation: ① For $j \leq w$, both C_j and $\overline{C_j}$ demonstrations exist. ② For $j > w$, at least $\overline{C_j}$ demonstrations are available. Thus, for missing $m-w$ labels, binary classification still provides $\overline{C_j}$-annotated demonstrations, mitigating label absence in incomplete retrieval datasets. Moreover, decomposing m-class classification into m binary tasks simplifies the learning problem. Determining presence/absence (e.g., "is this a cat?") is inherently easier than fine-grained discrimination among m alternatives (e.g., identifying specific animal species).

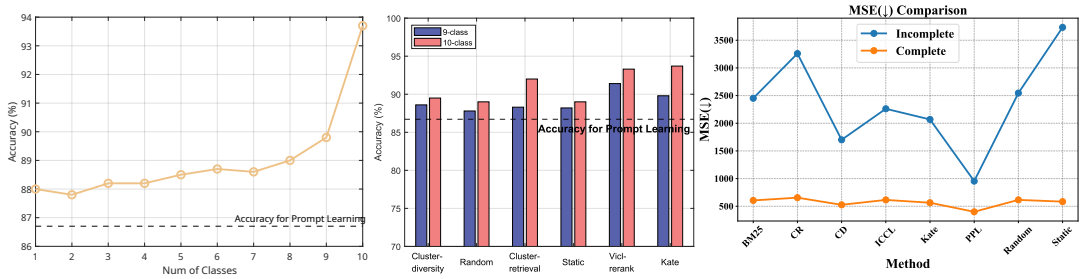
Table 1: The ground-truth label and other labels.

Ground-truth Label	Other Labels
Dog	Not cat, Not fish
Cat	Not dog, Not fish
Fish	Not dog, Not cat
C_j	$\overline{C_1}, \overline{C_2}, \cdots, \overline{C_{j-1}}, \overline{C_{j+1}}, \cdots, \overline{C_m}$
Fish	Not dog or cat

In *Integrated Prediction Stage*, we refine the test text's classification based on the test text and predictions from the *iterative judgments stage*. The final decision falls into one of three scenarios: ❶ **All Negative**: All predictions are negative ($\overline{C_1}, \overline{C_2}, \ldots, \overline{C_m}$). We directly perform m-label ICL classification. ❷ **Single Positive**: Exactly one positive prediction (e.g., $\overline{C_1}, \overline{C_2}, \ldots, C_j, \ldots, \overline{C_m}$). The data is classified as C_j. ❸ **Multiple Positive**: Multiple positive predictions (e.g., $\overline{C_1}, \overline{C_2}, \cdots, C_{j-1}, C_j, C_{j+1}, \cdots, \overline{C_m}$). We perform additional ICL classification among the positive labels (e.g., as a **three-class** task for three positives).

The training-free IJIP suffers from computational inefficiency for large m, producing lengthy prompts that raise token costs and even impair performance. To mitigate this, we propose aggregating fine-grained labels into coarse-grained categories, thereby reducing prompt length and token cost. For instance, as shown in Tab. 1, the label "Fish" can be represented as "Not dog or cat" (i.e., a non-mammal) instead of separate "Not cat" and "Not dog" labels. These coarse-grained labels are constructed by semantically clustering the original label set. Then, we develop a training-based IJIP variant with a cluster-number selector. This module adaptively selects the cluster number for per test text by predicting expected LLM performance across different cluster numbers.

The proposed IJIP framework is further extended to regression tasks with continuous scores and text generation tasks with infinite outputs. Regression and generation tasks inherently involve incomplete retrieval datasets, as their output spaces are infinite



(a) (b) (c)

Figure 3: Subfigure (a) shows the empirical study of IICL with different missing label numbers. Subfigure (b) shows the empirical study of IICL with different ICL methods in the classification task. Subfigure (c) shows the empirical study of IICL with different ICL methods in the regression task (translation quality assessment task).

and cannot be fully covered by any retrieval dataset. We also introduce a semi-supervised variant of IJIP to handle scenarios where only partial annotations are available. Comprehensive technical details are provided in Section 4.

We evaluate IJIP on six LLMs, seven tasks, and eight datasets, achieving considerable performance with a peak accuracy of 98.6% and a peak MSE of 0.62. Even when the incomplete retrieval database contains data for only a single label, IJIP maintains a peak accuracy of 89.2%. As a plug-and-play framework, IJIP can also be seamlessly integrated with zero-shot prompt learning and other ICL methods, yielding an average accuracy improvement 4.9%. Furthermore, under the standard ICL setting with complete retrieval datasets, IJIP achieves SOTA performance with an accuracy of 86.4%. Finally, we demonstrate the generality of IJIP by extending it to the image modality.

Our contributions are summarized as follows: ❶ We introduce the concepts of the *incomplete retrieval database* and *Incomplete In-context Learning (IICL)* and conduct an empirical study to assess their impact. Our findings reveal that IICL performance deteriorates as the number of missing labels increases. ❷ We propose the Iterative Judgments and Integrated Prediction (IJIP) method. By reformulating the IICL problem into a standard ICL scenario, IJIP effectively addresses the challenges posed by incomplete retrieval dataset and IICL. ❸ IJIP achieves SOTA performance across multiple datasets, tasks and LLMs. We also extend IICL to the *image modality* and generalize IJIP to zero-shot prompt learning and other ICL methods.

2 Problem Formulation

Definition 2.1 (Incomplete Retrieval Dataset). Let the label space of the test dataset be $\{C_1, C_2, \cdots, C_m\}$, and let the incomplete retrieval dataset be $\mathbf{D}_{\text{in}} = \{(x_i^{\text{in}}, y_i^{\text{in}})\}_{i=1}^{n}$. The dataset $\mathbf{D}_{\text{in}}$ contains data corresponding only to a subset of labels, with certain labels lacking associated data. Formally:

$$\begin{aligned} &y_i^{\text{in}} \in \{C_1^{\text{in}}, C_2^{\text{in}}, \ldots, C_w^{\text{in}}\}, \\ &\textit{s.t. } C_w^{\text{in}} \in \{C_1, C_2, \ldots, C_m\}, \text{ and } w < m, \end{aligned} \tag{1}$$

where $\{C_1^{\text{in}}, C_2^{\text{in}}, \cdots, C_w^{\text{in}}\}$ is label space of $\mathbf{D}_{\text{in}}$.

Definition 2.2 (Incomplete In-context Learning (IICL)). Let the k demonstrations retrieved from $\mathbf{D}_{\text{in}}$ be $\mathbf{D}_{\text{d}} = \{x_i^{\text{in}}, y_i^{\text{in}}\}_{i=1}^{k}$, test dataset be $\mathbf{D}_{\text{test}} = \{x_i\}_{i=1}^{h}$, and the LLMs with m classification task be denoted as f_{LLMs}^{m}, respectively. Given a test input x_i^{test} from $\mathbf{D}_{\text{t}}$, then IICL predicts the label

$$\hat{y}_i^{\text{t}} = f_{\text{LLMs}}^{m}(\mathbf{D}_{\text{d}}, x_i) \in \{C_1, C_2, \cdots, C_m\}. \tag{2}$$

2.1 Empirical Study of IICL

We evaluate the impact of incomplete retrieval dataset on IICL performance. Experiments employ CIFAR-10 (Abouelnaga et al., 2016) dataset and InternVL 2.5-8B (Chen et al., 2024) LLM. To simulate varying degrees of incompleteness, we progressively reduce the number of available labels from 10 to 1. Multiple ICL methods are evaluated—including Static, Random, Clustering-retrieval (Li and Qiu, 2023b), Kate (Liu et al., 2022), CD (Naik et al., 2023), and ICL-rerank (Zhou et al., 2024)—with 10 demonstrations per method. We further examine the effect of incomplete retrieval datasets in translation quality assessment, where the output space spans the continuous interval [0, 100]. Specifically, the retrieval set is restricted to examples with scores in [0, 20], and LLM performance is measured using Mean Squared Error (MSE).

(1) **IICL performance declines with increasing missing labels, yet remains superior to zero-shot prompting.** As shown in Subfigure (a) of

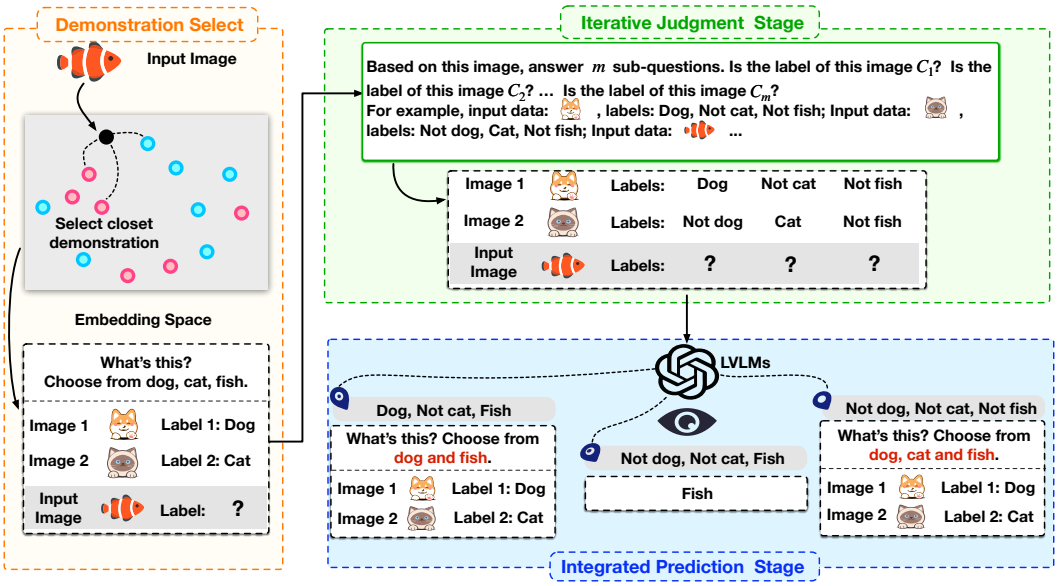


Figure 4: The overview of train-free *Iterative Judgments and Integrated Prediction (IJIP)*. Train-free IJIP retrieves the top k most similar data from the incomplete retrieval dataset as demonstrations based on their similarity to the test data. In the iterative judgment stage, train-free IJIP first queries the LLMs to classify the test data using the retrieved demonstrations. Then, train-free IJIP identifies labels receiving positive responses as candidate labels. If no labels receive positive judgments, train-free IJIP requests the LLMs to perform a full-label classification. Conversely, if multiple labels receive positive judgments, train-free IJIP initiates a second inquiry limited to these positively judged labels to refine the decision. Therefore, train-free IJIP queries LLMs at most twice and at least once, depending on the initial judgments.

Fig. 3, when data corresponding to more labels become missing, IICL performance gradually decreases. Notably, even with only one available label, IICL achieves 88.5% accuracy, outperforming the zero-shot prompt (86.7%). The removal of just one label causes a substantial accuracy drop from 93.7% to 88.5%. (2) **All methods experience accuracy degradation with missing labels.** Subfigure (b) of Fig. 3 shows that all six methods decline in the one-label-missing scenario, though all maintain superiority over the zero-shot baseline (86.7%). (3) **Complex tasks exhibit more severe performance degradation.** In translation quality evaluation (WMT EN-CS dataset, GLM4 9B), using an incomplete retrieval dataset leads to substantially higher MSE, indicating significantly poorer inference quality compared to classification tasks. **Conclusion:** Empirical results confirm that incomplete retrieval dataset significantly degrade IICL performance. This persistent performance gap highlights the critical need for developing missing-label robust methods.

3 Method

The IJIP framework includes two variants: training-free and training-based approaches. Both are designed to convert the IICL scenario with an incomplete retrieval dataset into a standard ICL with complete retrieval capability.

3.1 Train-free IJIP

3.1.1 Iterative Judgment Stage

Inspired by methods such as KATE (Liu et al., 2022), we retrieve the k most semantically similar labeled demonstrations from the incomplete retrieval dataset $\mathbf{D}_{\text{r}}$ for the test text. Specifically: ❶ *Vectorization*: Encode the test text x and each labeled text x_i^{in} from the retrival dataset $\mathbf{D}_{\text{in}}$ into vector representations using a pre-trained model f_{pre}, yielding $\mathbf{e}_x = f_{\text{pre}}(x)$, $\mathbf{e}_i = f_{\text{pre}}(x_i^{\text{in}})$, and the embedding set $\mathbf{E} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_n\}$. ❷ *Similarity Computation*: Compute the cosine similarity $s_i = \frac{\mathbf{e}_x \cdot \mathbf{e}_i}{\|\mathbf{e}_x\| \cdot \|\mathbf{e}_i\|}$ between x and each x_i^{in}, forming the similarity set $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$. ❸ *Demonstration Selection*: Select the top k labeled texts with the highest similarity scores, sort them in descending order, and form the demonstration dataset $\mathbf{D}_{\text{d}} = \{(x_i^{\text{d}}, y_i^{\text{d}})\}_{i=1}^{k}$.

As illustrated in Fig. 4, the m-class classification task is transformed into m binary sub-classification tasks. In the j-th sub-classification task, the objective is to determine whether the

label of the input data is C_j or $\overline{C_j}$. Consequently, the demonstration format is updated as $\mathbf{D}_{\text{d}}^{\text{new}} = \{x_i^{\text{d}}, y_{i,1}^{\text{d}}, y_{i,2}^{\text{d}}, \cdots, y_{i,m}^{\text{d}}\}_{i=1}^{k}$, where $y_{i,j}^{\text{d}} \in \{C_j, \overline{C_j}\}$ and $0 \leq j \leq m$. Then, the binary classification results $\hat{y}^1, \hat{y}^2, \cdots, \hat{y}^m$ are obtained using the sequential binary classifier f_{LLMs}^2. Formally:

$$\hat{y}^1, \cdots, \hat{y}^m = f_{\text{LLMs}}^2(\mathbf{D}_{\text{d}}^{\text{new}}, x), \text{ where } \hat{y}^j \in \{C_j, \overline{C_j}\}. \tag{3}$$

Remark: In the iterative judgment stage, we reformulate the m-class classification task into a series of binary classification tasks. Although this reformulation comprises m sub-questions, the **LLMs are queried only once during this stage.** Specifically, all m sub-questions and their corresponding retrieved demonstrations are integrated into a single consolidated prompt.

3.1.2 Integrated Prediction Stage

As the integrated prediction stage in Fig. 4 shows, the final classification integrates the test text x with binary predictions $\{\hat{y}^1, \hat{y}^2, \ldots, \hat{y}^m\}$ from Equation 3. Let $\mathbb{I}_x^j$ indicate whether $\hat{y}^j = C_j$, and $\mathbb{I}_x = \sum_{j=1}^{m} \mathbb{I}_x^j$ count the total positive predictions. The final prediction is then determined as:

$$\hat{y} = \begin{cases} f_{\text{LLMs}}^m(\mathbf{D}_{\text{d}}, x), & \text{if } \mathbb{I}_x = 0 \text{ or } m, \\ C_j, & \text{if } \mathbb{I}_x = 1 \text{ (where } \mathbb{I}_x^j = 1), \\ f_{\text{LLMs}}^u(\mathbf{D}_{\text{d}}, x), & \text{if } \mathbb{I}_x = u,\ 1 < u < m, \end{cases} \tag{4}$$

where $\mathbf{D}_{\text{d}}$ denotes demonstrations from Equation 2. When only one binary task predicts a positive label ($\mathbb{I}_x = 1$), that label C_j is directly assigned. For multiple positives ($\mathbb{I}_x = u > 1$), a u-class IICL selects among the candidate labels.

3.2 Train-based IJIP

The training-free IJIP faces a key limitation: its computational inefficiency with large class numbers m leads to prohibitively long prompts that both increase token costs and even degrade LLM performance. To address this, we introduce label aggregation, which groups fine-grained labels into coarse-grained labels to reduce prompt length and token cost. These coarse-grained labels are generated through semantic clustering of the original label set. A central challenge involves determining the optimal number of clusters. Assigning an XL-sized shirt to all individuals is clearly unreasonable; instead, shirt sizes should be adaptively selected based on individual measurements. Similarly, we argue that the number of clusters should be adaptively determined for each test text. Thus we develop a training-based IJIP variant equipped with a cluster-number selector. This module adaptively determines the suitable number of clusters for each test text by predicting LLM `performance score` across different cluster numbers. We construct (`text`, `in-context`, `label space`, `performance score`) tuples to train a deep learning model (cluster-number selector) with three inputs and one output (`performance score`). We select the label space with the greatest predicted performance score at each test text. The procedure comprises three stages: training dataset construction, model training, and cluster-number selection.

Training dataset construction: The training data for the cluster-number selector consists of (`text`, `in-context`, `label space`, `performance score`) tuples. ❶ `text` and `in-context` construction: We partition the incomplete retrieval dataset $\mathbf{D}_{\text{r}}$ into a text dataset $\mathbf{D}_{\text{t}}$ (20%) for training and a context dataset $\mathbf{D}_{\text{c}}$ (80%) for retrieval:

$$\mathbf{D}_{\text{r}} = \mathbf{D}_{\text{t}} \cup \mathbf{D}_{\text{c}},\ \mathbf{D}_{\text{t}} \cap \mathbf{D}_{\text{c}} = \varnothing,\ \{x_i^{\text{t}}, y_i^{\text{t}}\} \in \mathbf{D}_{\text{t}} \tag{5}$$

For each text $x_i^{\text{t}} \in \mathbf{D}_{\text{t}}$, corresponding in-context $\mathbf{D}_i^{\text{in,t}}$ is retrieved from $\mathbf{D}_{\text{c}}$ using Section 3.1.1's method. ❷ `label space` construction: For the `label space`, we define w as the maximum number of clusters. The label space undergoes K-means (Ullah et al., 2024) clustering with cluster counts varying from 2 to w. Denoting the original fine-grained label space as $\mathcal{Y}$, the 2-clustering label space is defined as $\mathcal{Y}^{(2)} = \{C_1^{(2)}, C_2^{(2)}\}$, where $C_1^{(2)}$ comprises all original fine-grained labels belonging to the first cluster (e.g., in Tab. 1, cluster label 1 may represent both "cat" and "dog"). Similarly, the l-clustering label space is expressed as $\mathcal{Y}^{(l)} = \{C_1^{(l)}, C_2^{(l)}, \ldots, C_l^{(l)}\}$. ❸ `performance score` construction To construct the `performance score`, we input the `text` x_i^{t}, its corresponding `in-context`, and `label spaces` with different cluster numbers into the LLM and employ the training-free IJIP to obtain the predicted label $\hat{y}_i^{\text{t}}$. The `performance score` is assigned a value of 1 if the predicted label matches the ground-truth label y_i^{t} of x_i^{t}, and otherwise it is recorded as 0. Formally:

$$\texttt{Performance Score} = \begin{cases} 1, & \hat{y}_i^{\text{t}} = y_i^{\text{t}}, \\ 0, & \hat{y}_i^{\text{t}} \neq y_i^{\text{t}}. \end{cases} \tag{6}$$

Model training: When training the cluster-number

selector $\mathcal{P}$, we adopt a classifier with a three-input, single-output architecture. The inputs correspond to the `text`, `in-context`, and `label space`, while the output is the `performance score`. Model architecture and training details are provided in Section B. **Cluster-number selection:** For a test text x, we retrieve its in-context using the method in Section 3.1.1 and generate candidate label spaces with varying cluster numbers via Section 3.2. These inputs are then processed by the classifier $\mathcal{P}$ obtained during model training. The label space achieving the highest predicted probability for predicted label 1 becomes the final label space for x, with its corresponding cluster number serving as the adaptively determined cluster number.

4 Scalability to Other Downstream Tasks, Other Scenario

❶ Building on the IICL, we introduce a **semi-supervised IICL** variant, referred to as *semi-supervised IICL*. In this setting, the incomplete retrieval dataset contains both labeled and unlabeled data, analogous to the semi-supervised learning paradigm. ❷ Beyond classification tasks, NLP also includes regression tasks with numerical scores and generation tasks with free-form text outputs. IJIP is extended to both scenarios. For **continuous score-based tasks**, the output range $[q, p]$ is discretized into u equal-width intervals—thereby transforming regression into classification—where $\Delta = (p - q)/u$. The i-th interval is:

$$\big[\, q + i\Delta,\ q + (i+1)\Delta \,\big], \quad \text{for } i = 0, 1, \ldots, u-1. \tag{7}$$

In training-free IJIP, u is a fixed hyperparameter, while training-based IJIP employs a neural network to adaptively determine u per test text with MSE as the `performance score`. For **text generation tasks**, we also adapt the two-stage IJIP methodology: first constructing a candidate set (Iterative Judgment Stage), then refining the prediction (Integrated Prediction Stage). Specifically, the LLM generates a top-k candidate set and selects the most appropriate response from it.

5 Experiments

5.1 Experimental Setup

Tasks, datasets and metrics. We evaluate IJIP on seven tasks: (1) image classification on CIFAR-10 (Krizhevsky et al., 2009) (Accuracy); (2) text classification on SST-5 (Socher et al., 2013) (Accuracy); (3) natural language inference (NLI) on SNLI (Bowman et al., 2015) (Accuracy); (4) semantic textual similarity (STS) on STS-B (Mean Squared Error MSE); (5) machine translation on WMT19 En-Zh (**?**) ; (6) text expansion on Gigatiny (ROUGE-1); and (7) text summarization on Gigaword (ROUGE-1). Tasks, datasets, and metrics are detailed in Sections J, K, and I, respectively. Lower MSE indicates better performance, while higher values are preferred for other metrics.

LLMs and baseline. For text-based tasks, we employ the following LLMs: GLM4 9B (**?**), LLaMA 3.1 8B (Touvron et al., 2024), and Qwen2.5 7B (Team, 2024). For image-based tasks, we utilize InternVL 2.5: 4B, 8B, and 26B. Baseline include BM25 (text only), Kate (Liu et al., 2022), Cluster-diversity (CD) (Naik et al., 2023), TTF (Liu et al., 2025a) (text only), DKNN (Li et al., 2025b), ICCL (Liu et al., 2024b), PPL (Gonen et al., 2023) (text only), and ICL-Rerank (Zhou et al., 2024) (image only).

Other setup. We employ CLIP (Radford et al., 2021b) as our pre-trained model. On CIFAR-10, SST-5, and SNLI, we consider two incomplete retrieval settings: single-label available and single-label missing. For STS-B, we use retrieval datasets with scores in $[0, 1]$ or $[0, 4]$. All ICL methods use 10 demonstrations. STS-B is divided into five intervals to align with the IJIP classification framework. In training-free IJIP, the maximum cluster count is 3, while for STS-B, the maximum interval number u in Equation 7 is 5. In generation tasks, the value of k in Top-k is set to 4.

5.2 Main Results

IJIP demonstrates strong performance across diverse tasks, model backbones, and incomplete-retrieval settings. As shown in Tab. 2, either the training-free or the training-based variant achieves the best result on every task, showing clear advantages over existing ICL baselines. Notably, even in the extreme setting where only a single label is available in the retrieval dataset, IJIP still attains a peak accuracy of 98.6%, a peak MSE of 0.65, a peak ROUGE-1 of 0.271, and a peak BLEU of 0.736. This trend suggests that IJIP is effective not only for classification, but also for regression and generation tasks. Overall, these results verify the effectiveness and robustness of IJIP in handling incomplete retrieval scenarios.

Table 2: Comparison of IJIP with other ICL methods. The best results are highlighted in bold. The second-best results are underlined. In generation tasks, only the training-free IJIP variant is employed.

Task & Data & Metric	Image classification: CAFIR-10 (Accuracy %↑)						NLI: SNLI (Accuracy %↑)					
Label Space	1 label			9 label			1 label			2 label		
LLMs	VL 4B	VL 8B	VL 26B	VL 4B	VL 8B	VL 26B	GLM	Llama	Qwen	GLM	Llama	Qwen
BM25	-	-	-	-	-	-	55.6	46.0	39.0	66.2	48.2	42.4
CD	78.7	86.8	95.2	83.2	88.3	96.3	67.5	33.6	83.2	79.6	55.2	88.0
ICCL	-	-	-	-	-	-	70.4	39.2	83.3	75.0	51.2	87.4
Kate	75.0	88.0	96.5	90.1	89.8	97.4	73.4	43.6	83.6	75.6	60.1	86.4
PPL	-	-	-	-	-	-	69.4	44.0	81.8	72.9	76.3	88.8
DKNN	77.4	84.8	91.9	81.2	85.9	92.1	74.7	38.0	79.6	71.2	68.4	81.7
TTF	76.5	86.4	93.3	84.4	87.9	94.0	54.1	32.8	58.6	58.4	56.0	64.3
ICL-rerank	74.0	88.1	96.1	90.8	88.2	96.9	-	-	-	-	-	-
Train-free IJIP	**88.5**	**89.2**	**97.4**	**93.9**	92.3	**98.6**	**75.6**	47.8	88.5	82.4	72.1	**92.5**
Train-based IJIP	87.7	89.0	96.3	92.4	**94.5**	97.8	74.5	**49.5**	**89.6**	**84.3**	**78.5**	90.4
Task & Data & Metric	Text classification: SST5 (Accuracy %↑)						STS: STS-B (MSE↓)					
Method	1 label			4 label			[0-1]			[0,4]		
LLMs	GLM	Llama	Qwen	GLM	Llama	Qwen	GLM	Llama	Qwen	GLM	Llama	Qwen
BM25	36.7	34.8	46.3	35.3	35.3	47.1	1.62	3.70	2.64	0.74	1.04	0.69
CD	42.3	34.4	43.9	47.3	41.6	51.1	1.07	4.87	3.79	0.68	1.43	0.89
ICCL	42.0	32.7	44.4	50.4	40.4	48.9	1.27	2.38	1.85	0.75	1.04	0.67
Kate	38.5	31.0	42.8	50.0	36.4	43.3	1.59	3.65	2.63	0.73	1.04	0.70
PPL	41.6	35.3	46.8	46.3	36.7	45.2	1.26	4.21	1.26	0.81	1.41	0.81
DKNN	39.9	33.4	41.9	39.0	34.1	43.5	-	-	-	-	-	-
TTF	35.7	31.4	38.1	36.2	36.2	48.3	-	-	-	-	-	-
Train-free IJIP	**51.3**	**49.2**	**51.7**	**54.1**	**53.6**	**56.0**	1.04	1.40	0.83	**0.65**	1.01	0.67
Train-based IJIP	49.7	48.4	49.9	51.4	51.7	53.2	**0.91**	**1.23**	**0.77**	0.66	**0.97**	**0.62**
Task & Data & Metric	Gigatiny (ROUGE-1↑)			Gigaword (ROUGE-1↑)			En-Zh (BLEU ↑)			Zh-En (BLEU ↑)		
LLMs	GLM	Llama	Qwen	GLM	Llama	Qwen	GLM	Llama	Qwen	GLM	Llama	Qwen
BM25	**0.132**	0.128	0.135	0.022	0.020	0.108	0.002	0.033	0.040	0.584	0.235	0.046
CD	0.100	0.102	0.240	0.023	0.025	0.134	0.005	0.134	0.250	0.675	0.250	0.461
Kate	0.122	0.125	0.253	0.041	0.032	0.106	0.051	0.137	0.104	0.696	0.253	0.518
DKNN	0.096	0.107	0.237	0.025	0.028	0.114	0.001	0.138	0.174	0.611	0.201	0.528
TTF	0.087	0.099	0.264	0.054	0.036	0.114	0.001	0.146	0.008	**0.755**	0.128	0.130
ICCL	0.116	0.115	0.230	0.034	0.032	0.106	0.002	0.139	0.111	0.387	0.193	0.069
PPL	0.074	0.110	0.249	0.056	0.037	0.115	0.002	0.144	0.165	0.680	0.184	0.500
Train-free IJIP	0.130	**0.133**	**0.271**	**0.067**	**0.040**	**0.138**	**0.068**	**0.223**	**0.254**	0.736	**0.286**	**0.638**

5.3 Ablation study

Ablating the Iterative Judgment stage causes substantial performance degradation. As shown in Figure 5, removing this component reduces average accuracy by 10.7% on CIFAR-10, 10.0% on SNLI, and 12.4% on SST-5, while increasing MSE by 0.87 on STSB.

Removing the Integrated Prediction stage also significantly harms performance. Since valid predictions cannot be obtained without this module on other datasets, evaluation is limited to CIFAR-10, SST-5, and SNLI. Subfigure (b) of Figure 6 shows that excluding Integrated Prediction lowers average accuracy by 13.5% on CIFAR-10, 13.2% on SST5, and 22.1% on SNLI.

6 Analysis and Discussion

6.1 Different Missing Label Proportions

We examine how IJIP performance correlates with missing label proportions. Using 10 demonstrations and progressively reducing label availability, results show accuracy declines with increased missing labels—for instance, on CIFAR-10 with InternVL 2.5-4B, accuracy drops from 92.1% to 88.5% (Fig. 6 subfigure (a)). Additional results are in Section E (Appendix).(2)

6.2 Expanding IJIP to Standard ICL, Semi-Supervised IICL, and Prompt Learning

We observe that the strategy of compressing the probabilistic output space to improve inference performance in IJIP can be extended to standard ICL, semi-supervised IICL, and prompt learning. Detailed procedures and corresponding experimental

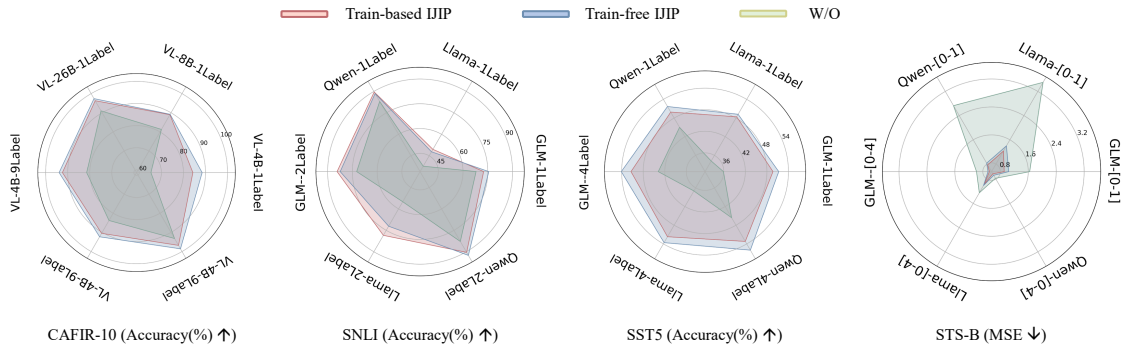
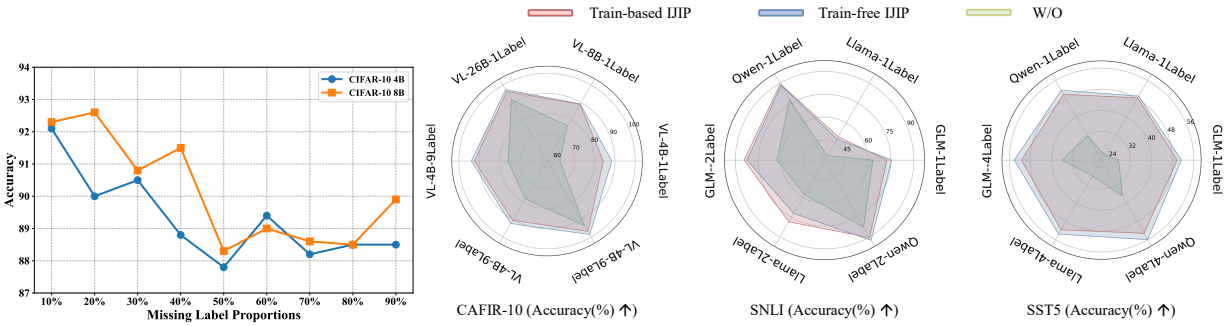


Figure 5: Results of IJIP with and without Iterative Judgments. "W/O" denotes without this component.



(a) (b) With and Without Integrated Prediction

Figure 6: Subfigures (a) The results of Train-free IJIP with different missing label proportions. and (b) The results of with and without Integrated Prediction

results are provided in Section G, Section F, and Section H, respectively.

6.3 Robustness Analysis

Variations in pre-trained models, and similarity metrics introduce stochastic rather than systematic effects on IJIP's performance. The detailed results are presented in Section C (Appendix). This demonstrates that IJIP is robust, as minor modifications to its components do not lead to significant performance degradation.

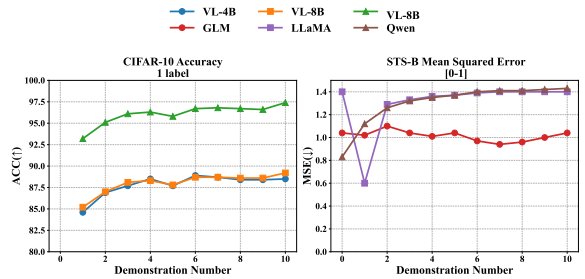


Figure 7: The results of Train-free IJIP with different demonstration numbers.

6.4 Non-Uniform Effect of Demonstration Number on IJIP Performance

We further study how the demonstration numbers affects IJIP. As shown in Fig. 7, its effect is task-dependent and non-uniform. On CIFAR-10, the accuracy improves noticeably when the demonstration numbers increases from a small value, but the improvement gradually saturates as more demonstrations are added. This suggests that additional demonstrations are most helpful when contextual information is insufficient, while their marginal benefit becomes limited beyond a moderate scale.

By contrast, on STS-B, the MSE exhibits clear fluctuations before stabilizing, rather than monotonically decreasing with more demonstrations. This implies that regression tasks are more sensitive to the specific composition of demonstrations, and that more demonstrations do not necessarily yield better predictions. Overall, these findings indicate that the optimal demonstration number should be chosen with respect to both task characteristics and efficiency considerations, instead of being increased indiscriminately.

Table 3: Average runtime (seconds per text) and average token usage (tokens per text) on SST5 dataset.

Method	BM25	CD	Kate	DKNN	TTF	ICCL	PPL	Train-free IJIP	Train-based IJIP	Train-free IJIP (5 shot)
Accuracy ↑	39.3	40.2	37.4	38.4	35.1	39.7	41.2	**50.7**	49.3	49.5
Token ↓	195.9	202.3	189.3	203.8	182.5	196.2	178.6	266.5	244.2	**174.3**
Time (s) ↓	1.95	0.26	0.27	0.22	0.27	0.36	0.32	0.43	0.49	0.38

6.5 Cost

IJIP requires two LLM queries per input, increasing computational costs. Fig. 7 shows that reducing demonstrations from 10 to 5 has minimal performance impact. Fewer demonstrations lower token usage and costs. Tab. 3 shows that on SST-5, reducing to 5 demonstrations decreases accuracy by only 1.2%, while cutting tokens to 174.3 and time to 0.38 s.

6.6 Performance Comparison Between IJIP and Zero-Shot Classification

In image classification, using CLIP models for similarity computation represents a commonly adopted zero-shot baseline, achieving 88.7% accuracy. As shown in Tab. 5, IJIP achieves higher accuracy compared to this zero-shot classification.

6.7 IJIP as a Plug-and-Play Enhancement for Existing ICL Methods

We further evaluate whether IJIP can improve existing ICL methods in a plug-and-play manner. Specifically, we augment CD and DKNN with training-free IJIP on SST5 using LLaMA 3.1 8B. As shown in Tab. 4, this integration consistently improves both methods across incomplete-label settings. For CD, the accuracy increases from 33.6% to 35.3% in the 1-label setting and from 55.2% to 62.8% in the 9-label setting, yielding an average gain of 4.6%. For DKNN, the corresponding improvements are from 38.0% to 41.7% and from 68.4% to 75.1%, with an average gain of 5.2%. These results show that IJIP is not limited to a standalone design, but can also complement existing retrieval-based ICL methods and further enhance their robustness under incomplete retrieval settings.

7 Conclusion

We introduce the concepts of incomplete retrieval dataset and IICL, empirically analyzing their effects and proposing IJIP as a solution. IJIP maintains strong performance even with 90% missing labels and can be adapted to various scenarios through task reformulation (Section 4). The framework extends to standard ICL, semi-supervised IICL, prompt learning, and includes a specialized variant for text generation, demonstrating broad versatility. **Related work, extended discussions, and more results are provided in the appendix.**

Table 4: Experimental Results of Training-Free IJIP as a Plug-and-Play Module. We augment CD and DKNN methods by incorporating training-free IJIP. Experiments are conducted using the LLaMA 3.1 8B LLM on the SST5 dataset.

	With Train-free IJIP		Without Train-free IJIP	
Method	1 Label	9 Label	1 Label	9 Label
CD	35.3	62.8	33.6	55.2
DKNN	41.7	75.1	38.0	68.4

Table 5: The results of zero-shot classification by Clip in CAFIR-10 dataset.

Clip	Train-free IJIP (9 Label)			Train-based IJIP (9 Label)		
Clip	VL 4B	VL 8B	VL 26B	VL 4B	VL 8B	VL 26B
88.7	93.9	92.3	98.6	92.4	94.5	97.8

Limitations

The training-free IJIP increases token usage, whereas the training-based variant requires additional training time, storage, and inference overhead. For instance, on SST5, training takes 20.3 minutes and 468 MB of storage. While these computational and storage overheads are non-negligible, they represent a worthwhile trade-off given the substantial performance gains achieved by IJIP.

Acknowledgments

This work was supported by the Shenzhen Science and Technology Program (Grant No. CJGJZD20240729141505007), the National Natural Science Foundation of China (Grant Nos. U2541229 and 62411540034), and the Ningbo Science and Technology Innovation 2025 Major Project (Grant No. 2025Z027).

References

Yehya Abouelnaga, Ola S Ali, Hager Rady, and Mohamed Moustafa. 2016. Cifar-10: Knn-based ensemble of classifiers. In *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 1192–1195. IEEE.

Farhad Akhbardeh, Cecilia Ovesdotter Alm, Marcos Zampieri, and Travis Desell. 2021. Handling extreme class imbalance in technical logbook datasets. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4034–4045.

Josh Attenberg and Foster Provost. 2010. Why label when you can search? alternatives to active learning for applying human resources to build classification models under extreme class imbalance. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 423–432.

Jack Bell, Luigi Quarantiello, Eric Nuertey Coleman, Lanpei Li, Mauro Madeddu, and Vincenzo Lomonaco. 2025. The future of continual learning in the era of foundation models: Three key directions. *arXiv preprint*.

Amanda Bertsch, Maor Ivgi, Emily Xiao, Uri Alon, Jonathan Berant, Matthew R. Gormley, and Graham Neubig. 2025. In-context learning with long-context models: An in-depth exploration. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Xusheng Cao, Haori Lu, Xialei Liu, and Ming-Ming Cheng. 2025. Class incremental learning for image classification with out-of-distribution task identification. *IEEE Transactions on Multimedia*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, and 1 others. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24185–24198.

Chen Cheng, Xinzhi Yu, Haodong Wen, Jinsong Sun, Guanzhang Yue, Yihao Zhang, and Zeming Wei. 2024. Exploring the robustness of in-context learning with noisy labels. *arXiv preprint arXiv:2404.18191*.

Jiawei Dai, Hanxiao Liu, Yizhou Wang, and Hongyu Wu. 2023. Promptagator: Fine-tuning retrieval models with contrastive learning for better in-context learning. *arXiv preprint arXiv:2304.05744*.

Alexander de Wynter. 2025. Is in-context learning learning? *arXiv preprint*.

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*, pages 4171–4186.

Benjamin Dherin. 2025. The implicit dynamics of in-context learning. *arXiv preprint*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

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 (Volume 1: Long Papers)*, pages 14014–14031.

Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2023. Complexity-based prompting for multi-step reasoning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Zhiwei Gao and 1 others. 2025. Knowledge memorization and rumination for pre-trained model-based class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2025*.

Manoj Garg and 1 others. 2022. Meta-learning for in-context learning tasks. *ICLR*. Available at `https://openreview.net/forum?id=Z2MavhD8bt`.

Hila Gonen, Srini Iyer, Terra Blevins, Noah A Smith, and Luke Zettlemoyer. 2023. Demystifying prompts in language models via perplexity estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10136–10148.

Karan Gupta, Sumegh Roychowdhury, Siva Rajesh Kasa, Santhosh Kumar Kasa, Anish Bhanushali, Nikhil Pattisapu, and Prasanna Srinivasa Murthy. 2023. How robust are llms to in-context majority label bias? *arXiv preprint arXiv:2312.16549*.

Guo Haixiang, Li Yijing, Jennifer Shang, Gu Mingyun, Huang Yuanyue, and Gong Bing. 2017. Learning from class-imbalanced data: Review of methods

and applications. *Expert systems with applications*, 73:220–239.

Zheng Yi Ho, Siyuan Liang, Sen Zhang, Yibing Zhan, and Dacheng Tao. 2024. Novo: Norm voting off hallucinations with attention heads in large language models. *arXiv preprint arXiv:2410.08970*.

Fanding Huang, Guanbo Huang, Xiao Fan, Yi He, Xiao Liang, Xiao Chen, Qinting Jiang, Faisal Nadeem Khan, Jingyan Jiang, and Zhi Wang. 2026. Semantic-space exploration and exploitation in rlvr for llm reasoning. *Preprint*, arXiv:2509.23808.

Fanding Huang, Jingyan Jiang, Qinting Jiang, Hebei Li, Faisal Nadeem Khan, and Zhi Wang. 2025. Cosmic: Clique-oriented semantic multi-space integration for robust clip test-time adaptation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9772–9781.

Quentin Jodelet and 1 others. 2025. Future-proofing class-incremental learning. *Machine Vision and Applications (Springer) / special issue 2025*.

David Minkwan Kim, Soeun Lee, and Byeongkeun Kang. 2025. Completely weakly supervised class-incremental learning for semantic segmentation. *arXiv preprint*.

Jannik Kossen, Yarin Gal, and Tom Rainforth. 2024. In-context learning learns label relationships but is not conventional learning. In *International Conference on Learning Representations*.

Alex Krizhevsky, Geoffrey Hinton, and 1 others. 2009. Learning multiple layers of features from tiny images.

Atsutoshi Kumagai, Tomoharu Iwata, Hiroshi Takahashi, Taishi Nishiyama, and Yasuhiro Fujiwara. 2025. Importance-weighted positive–unlabeled learning for distribution shift adaptation. In *Proceedings of the 2025 International Conference on Artificial Intelligence and Statistics (AISTATS 2025)*.

Vishnuprasadh Kumaravelu, P.K. Srijith, and Sunil Gupta. 2025. Evocl: Continual learning over evolving domains. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV 2025)*.

Chuyuan Li, Raymond Li, Thalia S. Field, and Giuseppe Carenini. 2025a. Delta-knn: Improving demonstration selection in in-context learning for alzheimer's disease detection. In *Proceedings of the 2025 Annual Meeting of the Association for Computational Linguistics (ACL 2025) — Long Papers*. ACL 2025; PDF available on ACL Anthology.

Chuyuan Li, Raymond Li, Thalia S Field, and Giuseppe Carenini. 2025b. Delta-knn: Improving demonstration selection in in-context learning for alzheimer's disease detection. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3873–3895.

Dongliang Li, Yuqiang Liu, Yichao Lu, and Wenxin Zhang. 2023. Contextualizing retrieval-augmented in-context learning: Diversity and retrieval. *arXiv preprint arXiv:2303.06334*.

Qiongxiu Li, Xiaoyu Luo, Yiyi Chen, and Johannes Bjerva. 2025c. Trustworthy machine learning via memorization and the granular long-tail: A survey on interactions, tradeoffs, and beyond. *arXiv preprint*. 2025 survey linking memorization and long-tail phenomena; long-tail .

Shuo Li, Fang Liu, Licheng Jiac, Lingling Li, Puhua Chen, Xu Liu, and Wenping Ma. 2025d. Prompt-based concept learning for few-shot class-incremental learning. *IEEE Transactions on Circuits and Systems for Video Technology*.

Tianle Li and 1 others. 2024a. Longiclbench: Long-context llms struggle with long in-context learning. *Arxiv Preprint*. Available at https://arxiv.org/abs/2402.04024.

Wen Li, Yihan Liu, Hao Chen, and Xin Zhang. 2025e. Explanation-based in-context demonstrations retrieval. *arXiv preprint*.

Xiao-Li Li, Philip S Yu, Bing Liu, and See-Kiong Ng. 2009. Positive unlabeled learning for data stream classification. In *Proceedings of the 2009 SIAM international conference on data mining*, pages 259–270. SIAM.

Xiaonan Li and Xipeng Qiu. 2023a. Finding support examples for in-context learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6219–6235.

Xiaonan Li and Xipeng Qiu. 2023b. Mot: Pre-thinking and recalling enable chatgpt to self-improve with memory-of-thoughts. *CoRR*.

Xiping Li, Xiangyu Dong, Xingyi Zhang, Kun Xie, Yuanhao Feng, Bo Wang, Guilin Li, Wuxiong Zeng, Xiujun Shu, and Sibo Wang. 2025f. Chi-square wavelet graph neural networks for heterogeneous graph anomaly detection. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 1565–1576.

Xiping Li and Jianghong Ma. 2026. Aim-cot: Active information-driven multimodal chain-of-thought for vision-language reasoning. *Preprint*, arXiv:2509.25699.

Xiping Li, Jianghong Ma, Kangzhe Liu, Shanshan Feng, Haijun Zhang, and Yutong Wang. 2024b. Category-based and popularity-guided video game recommendation: a balance-oriented framework. In *Proceedings of the ACM Web Conference 2024*, pages 3734–3744.

Xiping Li, Aier Yang, Jianghong Ma, Kangzhe Liu, Shanshan Feng, Haijun Zhang, and Yi Zhao. 2026. Cpgrec+: A balance-oriented framework for personalized video game recommendations. *ACM Transactions on Information Systems*, 44(3):1–44.

Ziqian Lin and Kangwook Lee. 2024. Dual operating modes of in-context learning. *ICLR*. Available at https://openreview.net/forum?id=HkAtRP9cOY.

Aishan Liu, Yuguang Zhou, Xianglong Liu, Tianyuan Zhang, Siyuan Liang, Jiakai Wang, Yanjun Pu, Tianlin Li, Junqi Zhang, Wenbo Zhou, and 1 others. 2024a. Compromising embodied agents with contextual backdoor attacks. *arXiv preprint arXiv:2408.02882*.

Hui Liu, Wenya Wang, Hao Sun, Chris Xing Tian, Chenqi Kong, Xin Dong, and Haoliang Li. 2025a. Unraveling the mechanics of learning-based demonstration selection for in-context learning. In *Association for Computational Linguistics 2025*.

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.

Wenzhuo Liu, Xin-Jian Wu, Fei Zhu, Ming-Ming Yu, Chuang Wang, and Cheng-Lin Liu. 2025b. Class incremental learning with self-supervised pretraining and prototype learning. *Pattern Recognition*, 157:110943.

Yinpeng Liu, Jiawei Liu, Xiang Shi, Qikai Cheng, Yong Huang, and Wei Lu. 2024b. Let's learn step by step: Enhancing in-context learning ability with curriculum learning. *arXiv preprint arXiv:2402.10738*.

Yu Liu and 1 others. 2025c. Sec-prompts: Semantic complementary prompting for few-shot class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2025*.

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.

Shuang Luo, Linjun Shou, Shuming Ma, Jiawei Liu, and Zhiwei Zhang. 2024. In-context learning with retrieval: A survey. *arXiv preprint arXiv:2401.06247*.

Xinxi Lyu, Sewon Min, Iz Beltagy, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Z-ICL: Zero-shot in-context learning with pseudo-demonstrations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2304–2317.

Farhad Mansouri, Xiaotian Liu, and Sanghyun Park. 2025. Learning from positive and unlabeled examples: Finite-sample complexity and bounds. *arXiv preprint*.

Lei Mao, Yuqiang Yang, Qi Wei, and Zhiwei Chen. 2024. A survey on data generation and task learning in in-context learning. *arXiv preprint arXiv:2403.06201*.

Daniel Mildenberger, Junjie Fang, and Yongming Xu. 2025. A tale of two classes: Adapting supervised contrastive strategies for imbalanced binary classification. In *Proceedings of NeurIPS 2025*.

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.

Saleh Momeni, Sahisnu Mazumder, Zixuan Ke, and Bing Liu. 2025. Inca: In-context continual learning assisted by an external continual learner. In *Proceedings of COLING 2025*. COLING 2025 — ICL"/".

Arman Nafar, Jun Chen, Daniel Khashabi, and Dan Roth. 2025. Learning vs retrieval: The role of in-context examples in large language models. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2025)*.

Ranjita Naik, Varun Chandrasekaran, Mert Yuksekgonul, Hamid Palangi, and Besmira Nushi. 2023. Diversity of thought improves reasoning abilities of large language models.

Milad Khademi Nori, Il-Min Kim, and Guanghui Wang. 2025. Autoencoder-based hybrid replay for class-incremental learning. *arXiv preprint*.

Ziqian Pan and 1 others. 2023. Dual operating modes of in-context learning. *ICML*, pages 2345–2356.

Guang Peng, Kai Zhang, and Weijia Zhao. 2025. Encode errors: Representational retrieval of in-context information. In *Findings of the Association for Computational Linguistics 2025*.

Kha Pham, Hung Le, Man Ngo, and Truyen Tran. 2025. Rapid selection and ordering of in-context demonstrations via prompt embedding clustering. In *Proceedings of the International Conference on Learning Representations (ICLR) 2025*. ICLR 2025 poster / paper; OpenReview PDF available.

Kiran Purohit, V. Venktesh, Sourangshu Bhattacharya, and Avishek Anand. 2025. Sample efficient demonstration selection for in-context learning. In *Proceedings of the 2025 International Conference on Machine Learning (ICML 2025)*. ICML 2025; preprint on arXiv:2506.08607.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021a. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International*

Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021b. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, and 1 others. 2018. Improving language understanding by generative pre-training.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020a. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020b. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Vasilii Rubin, Nikolaos Pappas, and Dimitris Christodoulou. 2022. Learning to retrieve: Towards a unified framework for information retrieval in pre-trained language models. *arXiv preprint arXiv:2203.02144*.

Ankit Singhal, Jayadeva, and Rajat K. De. 2014. On the selection of appropriate distances for gene expression data clustering. *BMC Bioinformatics*, 15(S2):S2.

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.

Qwen Team. 2024. Qwen2.5: A family of open-source large language models. `https://github.com/QwenLM/Qwen2.5`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, and Teven Le Scao. 2024. Llama 3: Open and efficient foundation language models. *arXiv preprint arXiv:2405.12345*.

Nasib Ullah, Erik Schultheis, Mike Lasby, Yani Ioannou, and Rohit Babbar. 2024. Navigating extremes: Dynamic sparsity in large output spaces. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*.

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.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022a. Emergent abilities of large language models. *Trans. Mach. Learn. Res.*, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022b. Chain-of-thought prompting elicits reasoning in large language models.

Jerry Wei, Le Hou, Andrew Kyle Lampinen, Xiangning Chen, Da Huang, Yi Tay, Xinyun Chen, Yifeng Lu, Denny Zhou, Tengyu Ma, and Quoc V Le. 2023a. 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.

Jerry Wei, Jason Wei, Yi Tay, Dustin Tran, Albert Webson, Yifeng Lu, Xinyun Chen, Hanxiao Liu, Da Huang, Denny Zhou, and 1 others. 2023b. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*.

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 *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1423–1436.

Susan Xie, Talia Nisan, Dale Schuurmans, and Richard Sutton. 2021. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*.

Kexin Yin. 2025. Which attention heads matter for in-context learning? *arXiv preprint*.

Zonghao Ying, Aishan Liu, Siyuan Liang, Lei Huang, Jinyang Guo, Wenbo Zhou, Xianglong Liu, and Dacheng Tao. 2026. Safebench: A safety evaluation framework for multimodal large language models. *International Journal of Computer Vision*, 134(1):18.

Shuyang Yu, Runxue Bao, Parminder Bhatia, Taha Kass-Hout, Jiayu Zhou, and Cao Xiao. 2025. Dynamic uncertainty ranking: Enhancing retrieval-augmented in-context learning for long-tail knowledge in llms. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8985–8997.

Jinghua Zhang, Li Liu, Olli Silvén, Matti Pietikäinen, and Dewen Hu. 2025. Few-shot class-incremental learning for classification and object detection: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yucheng Zhou, Xiang Li, Qianning Wang, and Jianbing Shen. 2024. Visual in-context learning for large vision-language models. *arXiv preprint arXiv:2402.11574*.

Yucheng Zhou, Tao Shen, Xiubo Geng, Chongyang Tao, Can Xu, Guodong Long, Binxing Jiao, and Daxin Jiang. 2023. Towards robust ranker for text retrieval. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 5387–5401. Association for Computational Linguistics.

Mingli Zhu, Siyuan Liang, and Baoyuan Wu. 2024. Breaking the false sense of security in backdoor defense through re-activation attack. *Advances in Neural Information Processing Systems*, 37:114928–114964.

Overview of the Appendix

This appendix includes our supplementary materials as follows:

- A review of related work in Section A.
- Details of the model architecture and training setup in Section B.
- Results of various pre-trained models and similarity computation in Section C.
- Descriptions of baseline methods in Section D.
- Analysis under different missing-label proportions in Section E.
- Experiments on semi-supervised IICL in Section F.
- Experiments on standard ICL in Section G.
- Experiments on prompt learning in Section H.
- More detailed descriptions of the evaluation metrics in Section I.
- Formal definitions of the tasks in Section J.
- More details about the dataset in Section K .
- Model access information and URLs in Section L.

A Related Work

In-context learning (ICL) (Bertsch et al., 2025; de Wynter, 2025; Dherin, 2025; Yin, 2025; Yu et al., 2025) has become a cornerstone technique in enabling large language models (LLMs) to perform a variety of downstream tasks using only a handful of example pairs, all without the need for gradient-based parameter updates. This inference-time adaptability has spurred a broad spectrum of studies examining key determinants of ICL effectiveness, such as the choice of in-context examples (Min et al., 2022; Cheng et al., 2024; Fei et al., 2023; Gupta et al., 2023; Li and Qiu, 2023a; Lyu et al., 2023; Wei et al., 2023b,a), their structural presentation (Min et al., 2022), sequencing (Lu et al., 2022; Wu et al., 2023), and associated labeling strategies (Wang et al., 2023). Recent innovations have increasingly incorporated retrieval-based mechanisms into the ICL framework, giving

rise to retrieval-augmented in-context learning (RetICL) (Luo et al., 2024), wherein demonstrations are dynamically sourced from external repositories to enrich task performance. Techniques employed for retrieval enhancement include embedding similarity measures (Rubin et al., 2022), diversity-promoting heuristics (Li et al., 2023), and retrievers fine-tuned via contrastive learning (Dai et al., 2023). A comprehensive overview by Luo et al. (Luo et al., 2024) underscores the efficacy of these strategies in bolstering both model robustness and generalization. Notably, prior studies commonly assume a shared label space between retrieved examples and target instances—an assumption often violated in practical scenarios such as domain adaptation or multilingual classification. To confront this overlooked challenge, we propose the Disjoint Output Spaces In-Context Learning (DOSICL) framework, designed to rigorously assess the capacity of LLMs to generalize across label-divergent source and target distributions. Building on similar principles, visual in-context learning (VICL) extends the paradigm to multimodal settings. The VICL methodology involves three components: retrieving relevant visual demonstrations, summarizing intent-specific visual information, and composing demonstrations tailored to the target query. Retrieval leverages both visual and textual modalities via pretrained encoders like ViT (Dosovitskiy et al., 2021; Bertsch et al., 2025; Li et al., 2024a), while relevance is further refined through cross-modal matching with models such as CLIP (Radford et al., 2021a; Pan et al., 2023; Lin and Lee, 2024), ensuring semantic and contextual alignment (Zhou et al., 2023). VICL capitalizes on the emergent abilities of LLMs to perform new tasks simply by conditioning on adapted prompts, without model retraining (Radford et al., 2018, 2019; Raffel et al., 2020a; Wei et al., 2022a,b; Fu et al., 2023; Xie et al., 2021; **?**; Mao et al., 2024; Garg et al., 2022; Li et al., 2025f, 2026; Li and Ma, 2026; Li et al., 2024b; Huang et al., 2026, 2025; Ying et al., 2026).

Class-Incremental Learning (CIL) (Zhang et al., 2025; Cao et al., 2025; Liu et al., 2025b; Li et al., 2025d; Jodelet et al., 2025; Nori et al., 2025; Kim et al., 2025; Liu et al., 2025c; Gao et al., 2025) enables models to learn new categories from sequential data while retaining knowledge of previous ones. The model initially learns from a subset of classes and adapts as new classes are introduced, without forgetting the earlier ones. CIL represents a scenario where the dataset does not include all potential categories during training. **Extreme Class-Imbalance Learning** occurs when class distributions are highly imbalanced, with the minority class containing very few or no samples (Attenberg and Provost, 2010; Akhbardeh et al., 2021). This situation is more severe than typical class imbalance problems, where the minority class is underrepresented but still present.

In summary, prior research on ICL and ICL has primarily focused on the selection of demonstrations and their integration into an appropriate context. However, these studies have largely overlooked potential anomalies in the context retrieval database. For example, in Class-Incremental Learning scenarios, the context retrieval database may lack data for newly introduced classes. Additionally, in cases of extreme class imbalance, certain labels may not have corresponding datasets. Furthermore, due to data labeling issues, some labels may lack annotated data altogether. These real-world challenges underscore the importance of exploring context learning in scenarios where data corresponding to specific labels is missing.

B Model Architecture and Training Details

Built upon the BERT-base-uncased architecture, our model incorporates five additional hidden layers. Training proceeds for 10 epochs with an initial learning rate of 1×10^{-3}, optimized via AdamW alongside a linear warmup scheduler to ensure stable convergence. To prevent overfitting, we apply dropout regularization at a rate of 0.2 within the classification layers.

C The Results of Different Pre-trained Models and Similarity Computation

To discuss the factors affecting IJIP's performance, we conduct a series of ablation studies focusing on critical design components: different pre-trained models, the method for computing semantic similarity, and the number and ordering of demonstrations. For the pre-trained model, we evaluate CLIP (Radford et al., 2021b), T5 (Raffel et al., 2020b), and BERT (Devlin et al., 2019). To compute semantic similarity, we compare Euclidean distance, cosine similarity, and Manhattan distance (Singhal et al., 2014). We vary the demonstration numbers from 1 to 10 and examine the impact of their ordering based on ascending and descending similarity.

As Figure 9 shows, employing cosine similarity, Euclidean distance, and Manhattan distance leads to average accuracies of 74.1%, 73.9%, and 73.8%, respectively, while the corresponding MSEs are 0.93, 0.90, and 0.95. *Variations in pre-trained models, similarity metrics, and the ordering of similarity scores appear to introduce stochastic rather than systematic effects on IJIP's performance.* As shown in Figure 8 and discussed, the average accuracies obtained using CLIP (Radford et al., 2021b), T5 (Raffel et al., 2020b), and BERT (Devlin et al., 2019) are 71.0%, 71.0%, and 71.5%, respectively. For the STS task, the MSE are 1.09, 1.07, and 1.10.

D Baselines

We perform experiments using various ICL and ICL methods, including Zero-shot Prompt, which is without in-context demonstrations; Static, where the top-k demonstrations are selected from the retrieval database; Random, where demonstrations are randomly chosen for each test input from the retrieval database; Clustering-retrieval (Li and Qiu, 2023b) organizes all demonstrations into k clusters, aiming to group similar examples, and subsequently chooses the most representative demonstration from each cluster, resulting in a final set of k demonstrations; Kate (Liu et al., 2022) identifies the most similar examples based on their sentence-level embeddings; and CD (Naik et al., 2023), where all demonstrations are clustered into k groups, and the demonstration closest to each cluster's center is selected to serve as the context demonstration. ICL-Rerank uses a 'Retrieval & Rerank' approach by first selecting the top-k most similar samples and then re-ranking them based on the image-text matching score with the query caption to improve relevance.

E The Results of Different Missing Label Proportions

Since the SNLI dataset comprises only three distinct labels, Tab. 2 already captures the effects of varying degrees of label omission on the experimental outcomes. Due to time limitations, the majority of our experiments are conducted using the CIFAR-10 dataset. To improve the reliability and generalizability of our results, we also incorporate the Fashion-MNIST dataset, a complementary ten-class image classification benchmark.

We investigate the relationship between the performance of IJIP and the proportion of missing labels. The experimental results are presented in Tab. 6. The accuracy of IJIP decreases as the proportion of missing labels increases, dropping from 77.4% to 47.5% on the Fashion-MNIST dataset using the InternVL 2.5-4B model.

Table 6: Performance of IJIP using InternVL 2.5-4B and InternVL 2.5-8B on CIFAR-10 and Fashion-MNIST under varying proportions of missing labels.

2*Class Nums	CIFAR10		Fashion MNIST	
	4B	8B	4B	8B
90%	88.5	89.9	47.5	52.8
80%	88.5	88.5	48.0	55.2
70%	88.2	88.6	50.0	59.0
60%	89.4	89.0	52.4	61.0
50%	87.8	88.3	56.4	63.8
40%	88.8	91.5	64.1	68.9
30%	90.5	90.8	67.9	72.7
20%	90.0	92.6	71.9	72.2
10%	92.1	92.3	77.4	78.9

F Semi-supervised IICL

The experimental setup for the semi-supervised IICL scenario is presented in Tab. 7, where IJIP continues to demonstrate considerable performance.

G Standard ICL

In the *Standard ICL* setting, where the retrieval dataset encompasses the complete set of target labels, our experiments are limited to the SST5, SNLI, and STS-B benchmarks due to time constraints. Within this aligned configuration, the proposed IJIP framework continues to leverage *Integrated Prediction Stage* to enhance inference accuracy. As shown in Tab. 8, IJIP demonstrates strong performance in the Standard ICL scenario, achieving a peak classification accuracy of 88.92% and a Mean Squared Error (MSE) of 0.62 on regression tasks, underscoring its robustness in contexts with consistent label semantics between demonstrations and queries.

H Prompt learning

In addition to context-based learning, prompt learning is a widely used method that leverages LVLMs. We integrate our method into prompt learning. The experimental results are provided in Figure 10. Our findings reveal that applying our method enhances the performance of prompt learning. Specifically,

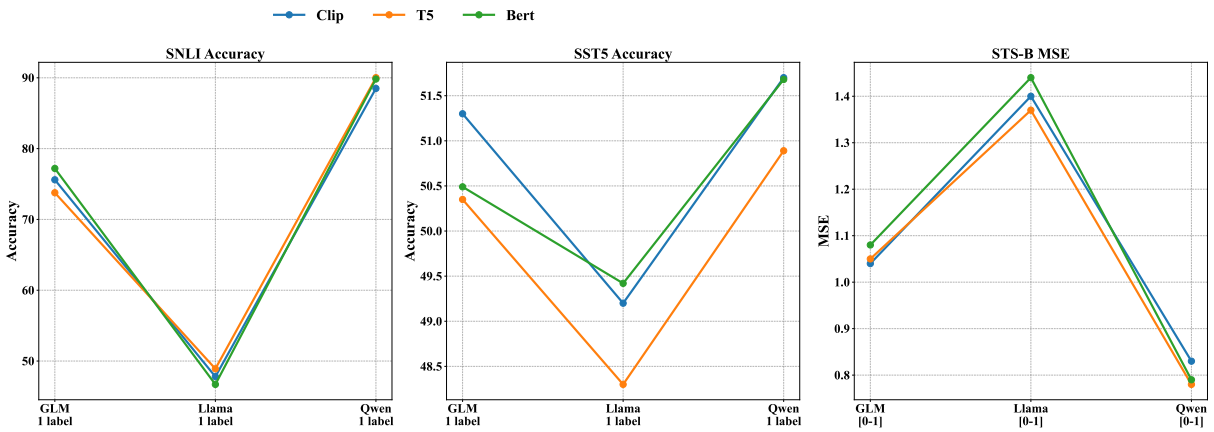


Figure 8: The performance of different pre-trained model.

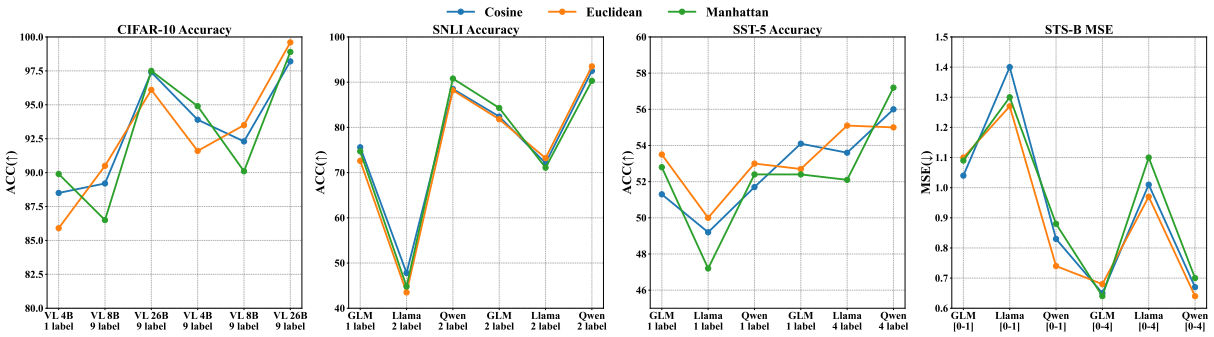


Figure 9: The performance of different similarity computation.

prompt learning experiences an average improvement of 4.8%. Due to time constraints, our experiments primarily focus on image classification tasks. To enhance the feasibility and diversity of the evaluation, we additionally incorporate three image classification datasets: Fashion-MNIST, WikiArt Genre, and WikiArt Artist.

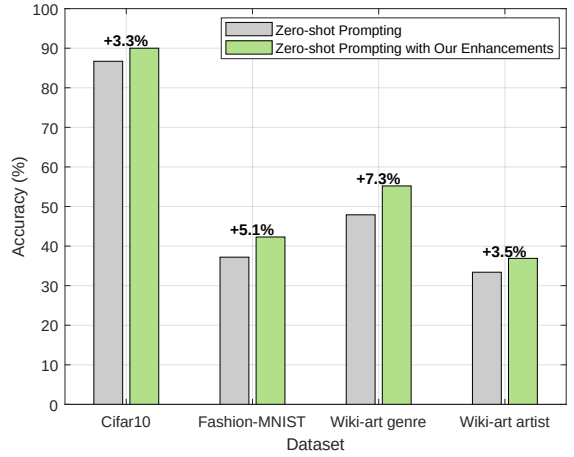


Figure 10: The effectiveness of IJIP in improving zero-shot prompt learning performance. Each experiment is conducted three times, and the average result is reported.

I Metrics

In this section, we provide detailed definitions and computational procedures for the four evaluation metrics adopted in our experiments, namely BLEU, ROUGE-1, Mean Squared Error (MSE), and Accuracy. These metrics are selected to cover different natural language processing tasks, including machine translation, summarization, semantic textual similarity, and text classification. Higher BLEU, ROUGE-1, and Accuracy scores indicate better performance, whereas lower MSE values correspond to stronger results.

I.1 BLEU

The Bilingual Evaluation Understudy (BLEU) score is a precision-oriented metric widely used in machine translation. It evaluates the overlap between a candidate sentence and one or more reference sentences based on n-gram matching. Specifically, BLEU is defined as

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right),$$

Table 7: The results of semi-supervised IICL scenario

	SNLI(Accuracy↑)					
	1 label			2 label		
	GLM4 9B	Llama3.1 8B	Qwen2.5 7B	GLM4 9B	Llama3.1 8B	Qwen2.5 7B
BM25	37.4%	38.4%	35.2%	58.8%	58.8%	57.3%
CD	80.3%	32.5%	87.6%	76.3%	76.3%	76.5%
Kate	76.0%	20.2%	80.4%	76.4%	76.4%	72.9%
DKNN	73.8%	20.0%	86.5%	72.3%	72.3%	73.9%
TTF	73.2%	23.8%	81.6%	73.0%	73.0%	73.0%
Train-free IJIP	82.9%	27.4%	86.4%	82.0%	79.6%	77.4%
	SST5(Accuracy↑)					
	1 label			4 label		
	GLM4 9B	Llama3.1 8B	Qwen2.5 7B	GLM4 9B	Llama3.1 8B	Qwen2.5 7B
BM25	31.7%	32.8%	45.3%	31.2%	39.4%	46.6%
CD	37.6%	35.1%	48.9%	45.9%	37.1%	43.9%
Kate	46.6%	30.5%	53.0%	45.9%	35.3%	50.2%
DKNN	46.2%	36.9%	48.2%	45.5%	40.7%	45.0%
TTF	36.7%	20.6%	43.4%	39.6%	47.7%	49.1%
Train-free IJIP	48.2%	32.9%	55.9%	46.1%	37.1%	53.4%

Table 8: The results of standard ICL scenario

	SNLI(Accuracy↑)			SST5(Accuracy↑)			STS-B(MSE↓)		
	GLM4 9B	Llama3.1 8B	Qwen2.5 7B	GLM4 9B	Llama3.1 8B	Qwen2.5 7B	GLM4 9B	Llama3.1 8B	Qwen2.5 7B
BM25	56.8%	45.8%	45.6%	34.2%	36.0%	46.8%	1.54	4.03	1.55
CD	76.4%	33.6%	84.3%	43.9%	36.4%	45.9%	0.85	**1.20**	0.99
Kate	75.1%	49.1%	83.6%	49.8%	35.8%	51.8%	0.66	1.28	**0.63**
DKNN	61.9%	28.1%	75.0%	43.0%	40.1%	42.8%	0.73	2.56	0.91
TTF	64.4%	20.4%	37.9%	43.4%	33.0%	43.9%	1.05	2.72	0.70
ICCL	75.6%	47.5%	83.7%	48.3%	37.0%	52.4%	0.67	1.11	0.69
PPL	74.9%	49.6%	81.1%	47.0%	36.3%	48.6%	0.65	1.76	0.68
Train-free IJIP	**77.6%**	**52.8%**	**88.9%**	**54.2%**	**43.2%**	**54.0%**	**0.62**	1.23	0.65

where p_n denotes the modified n-gram precision, w_n are weights (typically uniform), and BP is the brevity penalty introduced to penalize translations that are too short. A higher BLEU score reflects closer alignment with the reference translation.

I.2 ROUGE-1

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) is a family of metrics frequently applied to summarization and text generation tasks. ROUGE-1 measures the unigram-level overlap between the generated text and a reference. It is defined in terms of recall as

$$\text{ROUGE-1} = \frac{\sum_{\text{unigram} \in \text{Ref}} \min(\text{Count}_{\text{Gen}}(\text{unigram}), \text{Count}_{\text{Ref}}(\text{unigram}))}{\sum_{\text{unigram} \in \text{Ref}} \text{Count}_{\text{Ref}}(\text{unigram})}$$

where $\text{Count}_{\text{Gen}}$ and $\text{Count}_{\text{Ref}}$ denote the unigram counts in the generated and reference texts, respectively. ROUGE-1 thus captures the proportion of reference unigrams successfully covered by the system output.

I.3 Mean Squared Error (MSE)

Mean Squared Error (MSE) is employed to evaluate semantic textual similarity (STS) tasks by comparing predicted similarity scores against gold-standard labels. The metric is defined as

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2,$$

where y_i represents the ground-truth similarity score, $\hat{y}_i$ denotes the model prediction, and N is

the total number of samples. Smaller MSE values indicate higher fidelity of predicted scores to human-annotated similarities.

I.4 Accuracy

Accuracy is the most widely used evaluation metric in classification tasks, measuring the proportion of correct predictions. It is formally defined as

$$\text{Accuracy} = \frac{\text{\# correct predictions}}{\text{\# total samples}}.$$

A higher accuracy value implies that the model consistently assigns the correct label to input samples, thereby reflecting stronger classification performance.

J Task

This section of the appendix provides concise overviews of the benchmark tasks employed in our experiments, spanning both vision and language understanding domains. These tasks cover core capabilities such as classification, inference, similarity evaluation, summarization, and translation.

J.1 Image Classification

Image classification is a core task in computer vision that involves assigning a predefined category label to an entire image based on its visual content. It underpins a wide range of applications, such as medical diagnostics, autonomous driving, and product recognition.

J.2 Text Classification

Text classification is a cornerstone task in the field of Natural Language Processing (NLP), aimed at categorizing textual instances into a set of predefined labels. This task underpins a variety of real-world applications, including sentiment detection, spam filtering, intent classification, and topic identification. Performance is generally assessed using classification metrics such as accuracy, precision, recall, and the F1-score, depending on the specific task requirements.

J.3 Natural Language Inference

Natural Language Inference (NLI) is a foundational challenge in NLP that requires identifying the inferential relationship between two textual inputs: a *premise* and a *hypothesis*. The task involves assigning one of three relational labels: *entailment* (the hypothesis necessarily follows from the premise), *contradiction* (the hypothesis conflicts with the premise), or *neutral* (the truth of the hypothesis is indeterminate given the premise).

J.4 Semantic Textual Similarity

Semantic Textual Similarity (STS) evaluates the extent to which two textual segments convey equivalent meaning. Unlike classification-oriented tasks, STS is typically modeled as a regression problem, where the goal is to assign a real-valued score—commonly ranging from 0 (no semantic overlap) to 5 (identical meaning)—that reflects the semantic closeness of the input pair. STS is pivotal in a broad spectrum of downstream NLP applications, including paraphrase detection, information retrieval, question answering, and summarization. Accurate modeling of STS requires proficiency in understanding lexical semantics, syntactic constructs, contextual information, and often real-world knowledge.

J.5 Text Expansion

Text expansion refers to the task of extending a given short text fragment into a longer, semantically consistent, and contextually appropriate passage. This task can be seen as the inverse of summarization and serves applications such as dialogue generation, story completion, and content creation. It requires a model to preserve the semantic intent of the original input while generating additional coherent content, thereby testing the model's creativity and control in text generation.

J.6 Text Summarization

Text summarization involves generating a concise and coherent summary that captures the essential information of a source document while potentially using novel phrasing not present in the original text. Formally, given a document D, the goal is to produce a shorter sequence S that preserves the salient semantic content of D. Unlike extractive approaches that rely on selecting existing sentences, abstractive summarization requires the model to demonstrate generative linguistic capabilities and semantic abstraction.

J.7 Translation

Machine translation (MT) is the task of automatically converting text from a source language into a target language while preserving the original meaning and fluency. Formally, given a sequence $x = (x_1, \ldots, x_n)$ in the source language, the goal is to generate a target sequence $y = (y_1, \ldots, y_m)$

in another language. Translation remains one of the most established benchmarks for evaluating sequence-to-sequence models and has served as a cornerstone for advances in neural language modeling.

K Datasets

This section provides an overview of the benchmark datasets used in our experiments, covering both vision and natural language processing tasks. These datasets are employed to evaluate the model's performance across diverse domains and task types.

K.1 CIFAR-10

The CIFAR-10 dataset is a widely used small-scale image collection comprising 60,000 32x32 color images distributed across 10 categories, with 6,000 images per category. These images are divided into 50,000 training samples and 10,000 test samples. The dataset presents a range of challenges, including variations in angles, poses, lighting, and backgrounds, making it difficult for both machine learning and deep learning algorithms. Due to its manageable size and pixel value normalization during preprocessing, CIFAR-10 is commonly used for benchmarking image classification tasks in computer vision.

K.2 Stanford Sentiment Treebank - 5-way

The Stanford Sentiment Treebank (SST-5) is a widely-adopted benchmark for fine-grained sentiment classification. It extends the original binary SST dataset by providing human-annotated sentiment labels for phrases and sentences parsed from movie reviews. Each data instance is assigned one of five labels: "Very Negative, Negative, Neutral, Positive, or Very Positive". The task is framed as a single-sentence classification problem, making it a challenging test for models to discern subtle semantic differences and compositional meaning. Performance is measured using classification "Accuracy".

K.3 Stanford Natural Language Inference

The Stanford Natural Language Inference (SNLI) dataset (Bowman et al., 2015) stands as a foundational corpus in the study of natural language inference. It includes 570,000 sentence pairs, each consisting of a *premise* and a *hypothesis*, labeled as either *entailment*, *contradiction*, or *neutral*. The premises are derived from image captions in the Flickr30k dataset, while hypotheses were crafted and annotated by human annotators. SNLI's scale and linguistic richness have made it a cornerstone for training and evaluating deep learning models in textual entailment, particularly those focused on understanding relational semantics.

K.4 STS Benchmark

The Semantic Textual Similarity (STS) Benchmark is a canonical dataset for evaluating models on the task of Textual Similarity Estimation. It comprises sentence pairs drawn from multiple sources, including image captions, news headlines, and user forums. Each pair is annotated with a similarity score ranging from "0 (no semantic relation) to 5 (semantic equivalence)". The STS Benchmark provides a standardized training, development, and test split, enabling fair comparison across different models. The standard evaluation metric is the Pearson correlation between predicted and gold scores, while "Mean Squared Error (MSE)" is also commonly reported to quantify prediction errors.

K.5 STS 2014, 2015, 2016

The STS datasets from the SemEval 2014, 2015, and 2016 shared tasks on Semantic Textual Similarity are collections of test sets used for evaluation. Unlike the STS Benchmark, these datasets are typically used "exclusively for testing", as they do not have official training splits. They contain sentence pairs from diverse domains such as news headlines, forum discussions, and glossaries. Each pair is annotated with a gold similarity score. These datasets are instrumental for evaluating the cross-domain generalization and robustness of models trained on other STS data (e.g., STSB). Performance is evaluated using the "Pearson correlation" coefficient.

K.6 Gigaword

The Gigaword dataset is a large-scale summarization corpus extensively used for training and evaluating models on "Abstractive Summarization". It consists of over 4 million news articles and their corresponding headlines from various news sources (e.g., Reuters, AFP). The standard task is to generate a concise headline given the first sentence (or the first few sentences) of the article. This task tests a model's ability to identify the most salient information and compress it into a very short summary. Model performance is primarily evaluated using the "ROUGE" metric suite, particularly "ROUGE-1" F1 score.

K.7 Gigatiny

Gigatiny is a curated and reduced version of the massive Gigaword dataset, designed to facilitate faster experimentation and prototyping in summarization research. It retains the same structure and objective as its parent dataset—generating a headline from an article's first sentence—but contains a significantly smaller subset of examples. This makes it a valuable resource for efficient hyperparameter tuning and initial model testing before scaling to the full Gigaword corpus. Evaluation is likewise performed using "ROUGE" metrics.

K.8 WMT19 Chinese-English Validation Set

The wmt19-valid-only-zh_en dataset, serves as a validation set for Chinese-to-English (zh→en) translation within the WMT19 news translation domain. It contains 3,981 parallel sentence pairs, where each entry provides a Chinese source sentence and its corresponding English reference translation. This dataset focuses on news-related content and is designed primarily for model validation and performance benchmarking rather than large-scale training. Translation quality on this benchmark is typically evaluated using automatic metrics such as the "BLEU" score against the provided human references.

K.9 WMT19 English-Chinese Validation Set

To obtain an English-to-Chinese (en→zh) validation set, we reversed the language direction of the wmt19-valid-only-zh_en dataset by swapping the source and target fields in each parallel sentence pair. This process preserves the original sentence alignment and translation quality while converting the translation direction from Chinese-to-English to English-to-Chinese. The resulting dataset therefore shares the same content and size (3,981 sentence pairs) as the original validation set but provides a complementary benchmark for evaluating English-to-Chinese translation models under the WMT19 news domain.

L THE URL OF Models

Table 9: NLP Tasks, Datasets, and Their URLs

Task	Dataset	URL
Image Classification	CIFAR-10	https://huggingface.co/datasets/uoft-cs/cifar10
Text Classification	SST5	https://huggingface.co/datasets/SetFit/sst5
Natural Language Inference	SNLI	https://huggingface.co/datasets/stanfordnlp/snli
Semantic Textual Similarity	STS14	https://huggingface.co/datasets/mteb/sts14-sts
	STS15	https://huggingface.co/datasets/mteb/sts15-sts
	STS16	https://huggingface.co/datasets/mteb/sts16-sts
	STSB	https://huggingface.co/datasets/SetFit/stsb
Text Summarization / Text Expansion	gigaword	https://huggingface.co/datasets/Gabriel/gigaword_swe
	gigatiny	https://huggingface.co/datasets/SpeedOfMagic/gigaword_tiny
Translation	WMT19 En–Zh	https://huggingface.co/datasets/WillHeld/wmt19-valid-only-zh_en

Table 10: Large Language Models and Their URLs

Model	URL
LLAMA-3.1-8b	https://huggingface.co/meta-llama/Llama-3.1-8B
Qwen2.5 7b	https://huggingface.co/Qwen/Qwen2.5-7B
GLM4 9B	https://huggingface.co/zai-org/glm-4-9b
LLAMA-3.2-3b	https://huggingface.co/meta-llama/Llama-3.2-3B
GPT-4o	https://platform.openai.com/docs/models/gpt-4o