TACO: Enhancing Multimodal In-context Learning via Task Mapping-Guided Sequence Configuration

Yanshu Li[1], Jianjiang Yang[2], Tian Yun[1], Pinyuan Feng[3], Jinfa Huang[4], Ruixiang Tang[5]*

[1]Brown University, [2]University of Bristol, [3]Columbia University,
[4]University of Rochester, [5]Rutgers University
`yanshu_li1@brown.edu, ruixiang.tang@rutgers.edu`

Abstract

Multimodal in-context learning (ICL) has emerged as a key mechanism for harnessing the capabilities of large vision–language models (LVLMs). However, its effectiveness remains highly sensitive to the quality of input ICL sequences, particularly for tasks involving complex reasoning or open-ended generation. A major limitation is our limited understanding of how LVLMs actually exploit these sequences during inference. To bridge this gap, we systematically interpret multimodal ICL through the lens of **task mapping**, which reveals how local and global relationships within and among demonstrations guide model reasoning. Building on this insight, we present **TACO**, a lightweight transformer-based model equipped with task-aware attention that dynamically configures ICL sequences. By injecting task-mapping signals into the autoregressive decoding process, TACO creates a bidirectional synergy between sequence construction and task reasoning. Experiments on five LVLMs and nine datasets demonstrate that TACO consistently surpasses baselines across diverse ICL tasks. These results position task mapping as a novel and valuable perspective for interpreting and improving multimodal ICL.

1 Introduction

In-context learning (ICL) is a paradigm in which models make predictions by conditioning on a sequence of input–output demonstrations, without updating their parameters. This approach enables models to rapidly adapt to new tasks using only a few examples provided at inference time. Initially, ICL gained significant traction in the domain of large language models (LLMs), where it has demonstrated impressive performance across a wide range of tasks (Olsson et al., 2022; Garg et al., 2023). More recently, the concept has been

extended to the multimodal setting, giving rise to multimodal ICL, where large vision–language models (LVLMs) learn from interleaved image–text sequences and support complex multi-image reasoning. This capability has become a cornerstone of modern LVLMs (Alayrac et al., 2022; Chen et al., 2024c; Bai et al., 2025), enabling more flexible and generalizable multimodal understanding.

Despite significant progress in multimodal ICL, configuring effective input sequences remains an open challenge. A standard ICL sequence consists of an instruction, a set of in-context demonstrations (ICDs), and a query sample (see Figure 1). Prior studies have shown that even small changes to these sequences can substantially alter LVLM predictions (Schwettmann et al., 2023; Zhou et al., 2024; Li et al., 2025c). These findings highlight the need for robust configuration strategies. However, as multimodal ICL involves diverse cross-modal interactions, our mechanistic understanding is still limited. As a result, existing methods depend on hand-crafted metrics to assess each ICD's contribution to LVLM reasoning (Iter et al., 2023; Fan et al., 2024). Rather than relying on specific metrics, we propose a more effective model-centric alternative. Specifically, we address two research questions:

How do multimodal sequences influence the ICL performance of LVLMs? (§2) We introduce **task mapping** as a new lens for understanding how input sequences drive multimodal ICL. In the model's latent space, each ICD defines a **local** task mapping from its modalities to its output, and these are synthesized into a **global** task mapping that produces the query response. To investigate this, we develop a probing framework that measures how LVLMs exploit these mappings across sequences. Our study yields two insights: (1) task mapping is essential for effective multimodal ICL, as it guides the alignment between input–output patterns across ICDs and the query; and (2) LVLMs perform better when ICDs form a cohesive task structure, espe-
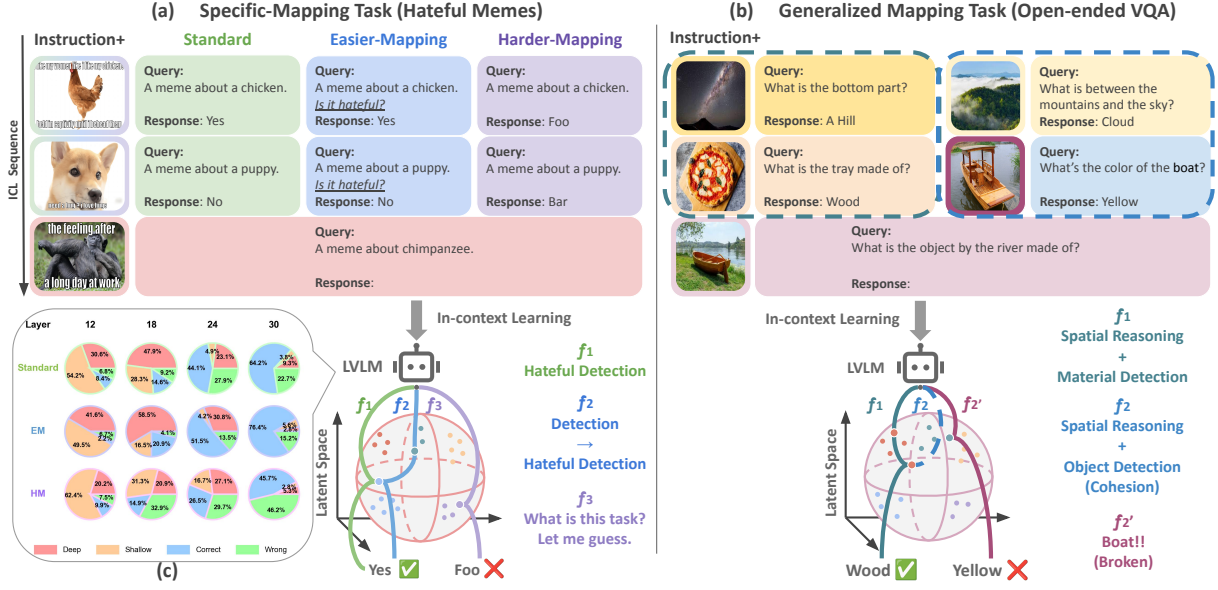
---

*Corresponding author.

Figure 1: Examples of 2-shot multimodal ICL. (a) In specific-mapping tasks, each ICD's local mappings are relatively consistent, and the ICL sequence's global mapping matches them. Their clarity directly affects the LVLM's reasoning process. The in-context lens in (c) also reflects this latent reasoning shift induced by task mapping. (b) In generalized-mapping tasks, LVLM needs to integrate each local mapping into a cohesive global mapping for reasoning. Overreliance on isolated features (e.g., the visual cue of a boat) can break this cohesion.

cially in complex cross-modal scenarios. These findings establish task mapping as a principled lens for analyzing and enhancing multimodal ICL.

How can we enhance the ICL sequence configuration for effective task mapping? (§3) Grounded in our theoretical analysis of task mapping, we propose **TACO** (Task-Aware model for in-COntext Learning), a lightweight transformer-based model that explicitly incorporates task mapping into the selection of ICDs. TACO first encodes the query and instruction to infer task intent, then retrieves ICDs that are both semantically relevant and aligned in reasoning steps. A specialized attention mechanism highlights ICDs that support a coherent interpretation, and layer-wise refinement lets ICDs reinforce one another, producing sequences that enable consistent task reasoning. This task-aware configuration significantly improves the robustness and accuracy of multimodal ICL. Extensive experiments with five advanced LVLMs and nine datasets demonstrate TACO's superior performance, validating its effectiveness and generality.

Our main contributions can be summarized in three-fold:

- To fill the gap in research on multimodal ICL mechanisms in LVLMs, we propose a task mapping framework that systematically captures task distributions across ICDs within an

ICL sequence. This framework not only offers a unified view of multimodal ICL under diverse scenarios but also sheds light on the internal behavior of LVLMs during ICL.

- Within this task-mapping framework, we propose TACO, a lightweight transformer-based model designed to adaptively retrieve and arrange ICDs from a dataset, yielding optimal ICL sequences for a target LVLM. Evaluations on five LVLMs and nine benchmarks show that TACO achieves superior performance over prior configuration strategies.

- We carry out an extensive analysis and ablation study of TACO, isolating the role of each component and design decision. The results provide insights into TACO's underlying mechanisms and further demonstrate the effectiveness of our task mapping framework for ICL enhancements and applications.

2 Task Mapping in Multimodal ICL

In this section, we focus on exploring task mapping in ICL. We first define task mapping (§2.1) and, through systematic empirical experiments, evaluate its impact on multimodal ICL (§2.2) and uncover how LVLMs leverage task mapping across the entire ICL sequence (§2.3). All experiments in this section are conducted on two LVLMs:

OpenFlamingov2-9B (Awadalla et al., 2023) and Idefics2-8B (Laurençon et al., 2024). Results are reported as the average across both models.

2.1 Identifying Task Mapping in multimodal ICL

Notations. In this work, we mainly focus on ICL for image-to-text tasks, where ICL sequences are organized in an interleaved image-text format. Toward a unified template for various tasks, we reformat ICDs as triplets (I, Q, R), where I is an image, Q is a task-specific text query, and R is the ground-truth response. The query sample is denoted as $(\hat{I}, \hat{Q})$. Formally, ICL can be represented as:

$$\hat{R} \leftarrow \mathcal{M}(S^n) = \mathcal{M}(Inst; \underbrace{(I_1, Q_1, R_1), ..., (I_n, Q_n, R_n)}_{n \times ICDs}; (\hat{I}, \hat{Q})),$$
(1)

where $\mathcal{M}$ is a pretrained LVLM, S^n is an ICL sequence consisting of an instruction $Inst$, n-shot ICDs and a query sample, as shown in Figure 1.

Task Mapping Definition. We define **task mapping** as a model-learnable inferential process that transforms input modalities into their outputs within the LVLM's latent space, capturing both local and global relationships in ICL. Each ICD (I_i, Q_i, R_i) possesses a local task mapping f:

$$f_i : (I_i, Q_i) \rightarrow R_i, i = 1, 2, ..., n.$$
(2)

specifying how its image and query jointly map to the response. Then LVLM's generation on the target query sample $(\hat{I}, \hat{Q})$ can be viewed as a global task mapping $\hat{f}$:

$$\hat{f} : (\hat{I}, \hat{Q}) \rightarrow \hat{R}.$$
(3)

Task mapping is inherently indeterminate and complex. To enable systematic analysis, we first consider a scenario where all local mappings f_i are nearly identical and coincide with the query's target mapping. This setup is common for tasks that require the LVLM to follow a fixed reasoning path. We term these as **specific-mapping tasks**. In such tasks, I, Q, and R often exhibit structural consistency, which facilitates component-level analysis.

Visualization. We employ a specific-mapping task, HatefulMemes (Kiela et al., 2020), to reveal task mapping. Here, each local mapping f_i is defined by a binary classification: given a meme image with its caption, determine whether it contains harmful content and output "yes" or "no." (Figure 1(a)) We use the validation split as our query set

and sample n ICDs from the training split using Random Sampling (RS) with a normal distribution to configure the ICL sequences. To highlight f_i, we create two setups:

- Easier Mapping (*EM*): Augment Q_i with an explicit task hint "Is it hateful?".
- Harder Mapping (*HM*): Replace R_i (yes/no) with non-semantic words foo/bar.

To explore how task mapping influence LVLM inference, we introduce in-context lens based on logit lens. It defines four anchor word categories: "Shallow" for superficial task recognition, "Deep" for deeper recognition, "Correct" for the query's true answer, and "Wrong" for its opposite (details in Appendix C.2). Figure 1(c) visualizes the evolution of internal token outputs under varying task mappings, illustrating the model's reasoning process. It shows that *EM* greatly enhances the model's ability for deeper task recognition, while *HM* leads to a persistent lack of task awareness, causing the model to rely on random guessing.

2.2 Task Mapping is Key to Multimodal ICL

To further examine task mapping's role in multimodal ICL under specific-mapping tasks, we isolate the individual contributions of labels (i.e., "yes" vs. "no"), image features, and task mappings to LVLM performance on HatefulMemes.

Setups. We introduce targeted ablation settings that selectively impair label reliability and visual clarity, allowing an evaluation of whether LVLMs primarily rely on task mapping over these isolated factors. Specifically, we define: 1. Wrong Labels (**WL**): Invert 75% R_i labels (yes↔no). 2. Blurred Images (**BI**): Applying Gaussian blur to all I_i. We also apply *EM* solely to $\hat{Q}$, denoted as *EM($\hat{Q}$)*. *BI($\hat{I}$)* refers to applying *BI* solely to $\hat{I}$. The details for these settings are provided in Appendix C.4.

Results. Figure 2 shows the LVLM's ICL performance on the same sequences under different settings. The findings are as follows:

Better capturing task mapping consistently improves performance. As shown in Figure 2(a), across all shot counts, *EM > Standard > HM* in a clear descending order. This aligns with our observations from in-context lens. In Figure 2(b), removing instructions, which serve as higher-level guidance enabling LVLMs to more deeply capture and utilize f_i, generally lowers performance. Yet "*EM* w/o $Inst$" still surpasses *Standard*.

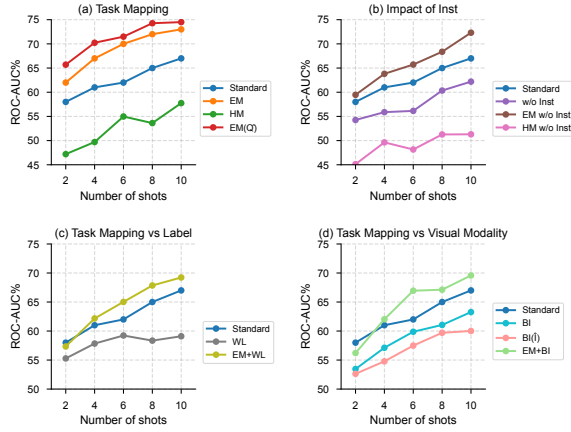**Query sample is pivotal.** Surprisingly, Figures

Figure 2: Results on HatefulMemes under various settings. "+" denotes combining two settings.
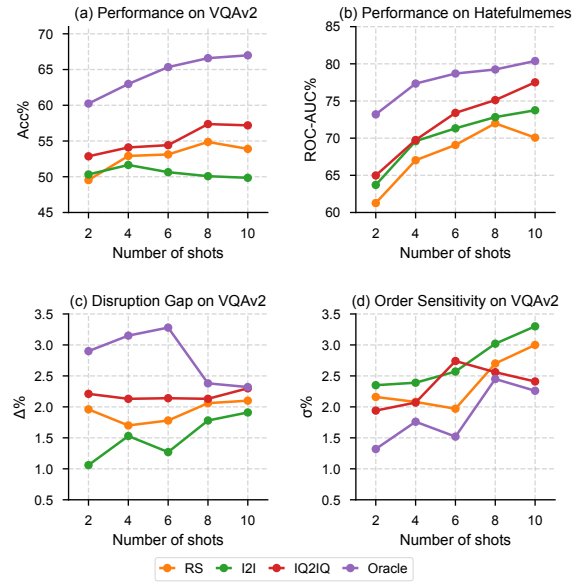

Figure 3: (a-b) Results of different ICL sequence configuration methods on VQAv2 and HatefulMemes. (c-d) Task mapping cohesion analysis of different ICL sequence configuration methods on VQAv2.

2(a) and (d) show that modifying $\hat{I}$ or $\hat{Q}$ causes greater performance variations than altering all ICDs. We hypothesize that LVLMs prioritize analyzing the query sample and use pretrained knowledge to constrain global task mapping accordingly.

Task mapping outweighs labels and visual cues. In the **WL** setting, performance drops (Figure 2c), yet stronger task mappings fully recover it. Likewise, in the **BI** setting, the loss is completely offset by enhanced mappings (Figure 2d). This suggests that both labels and visual modality affect multimodal ICL, but better utilization of task mapping can yield significant performance gains to address deficiencies in unimodal information.

2.3 ICL Needs Cohesive Global Mapping

Building on the central role of task mapping in multimodal ICL, we introduce **generalized-mapping tasks** to capture real-world challenges in which f_i exhibits nuanced or broad variability. Unlike specific mapping tasks, which represent a special case, they involve greater diversity in Q_i and R_i, making component-level manipulation difficult. We therefore adopt a sequence-level analysis, illustrated on the open-ended VQAv2 task (Goyal et al., 2017).

Setups. Three sequence configuration methods are evaluated: Random Sampling (RS), similarity-based retrieval, and an idealized Oracle. In similarity-based retrieval, ICDs are selected by CLIP cosine similarity, using either image-only alignment (I2I) or joint image query alignment (IQ2IQ). The Oracle method greedily chooses each ICD to maximize the log likelihood of generating the ground-truth response while accounting for the cumulative influence of prior demonstrations (com-

putational details in Appendix C.3).

Hypothesis. Figure 3(a) and (b) show that multimodal alignment via IQ2IQ consistently outperforms unimodal alignment (I2I) and RS on both datasets. Meanwhile, Oracle consistently achieves the highest accuracy. An unexpected finding is that I2I performs worse than RS on VQAv2 but not on HatefulMemes. We hypothesize that **task mapping cohesion** explains this phenomenon. In generalized-mapping tasks, effective ICL requires ICDs to jointly support complex reasoning. When performing ICL with certain sequences configured via I2I, isolated feature matching introduces a fragmented reasoning bias that leaves the sequence's global mapping cohesion **broken**.

Proof. To validate this hypothesis, we introduce two metrics for measuring task mapping cohesion: Disruption Gap (Δ) and Order Sensitivity (σ) (details in Appendix C.5). These metrics reflect the impact of cohesive task mapping on multimodal ICL, with higher Δ and lower σ indicating stronger reliance on cohesive task mapping. Figure 3(c-d) shows that Oracle achieves the highest Δ and lowest σ across all shots, proving its ability to construct cohesive sequences through holistic consideration of preceding ICDs. However, as shots increase to 8 and 10, Oracle's Δ surges while σ plunges, revealing potential local optimization issues and accumulated bias in longer sequences. Meanwhile, I2I con-

sistently underperforms RS on both metrics, while IQ2IQ surpasses RS but remains unstable, aligning with accuracy trends in generalized-mapping tasks and supporting our hypothesis.

Finally, based on performance, Δ and σ, we can categorize all experimental ICL sequences into four types for case studies (see Appendix C.6): (1)-(2) sequences impaired by isolated dependencies (e.g., similar image features and local mapping bias), (3) sequences resembling specific-mapping tasks, and (4) the most common type, featuring diverse local mappings that collectively enhance cohesive task mapping. Such diversity enables LVLMs to overcome excessive reliance on superficial features and achieve superior multimodal ICL performance.

3 The Proposed Method: TACO

Motivation. Task mapping plays a crucial role in enabling effective ICL for LVLMs, as discussed in §2. Thus, to construct high-quality ICL sequences, two objectives must be met: (1) each ICD should contribute a meaningful local mapping that supports the reasoning process, and (2) the sequence as a whole should form a cohesive global mapping that aligns with the target task. These objectives reflect how models implicitly organize and utilize contextual information during inference. However, existing metric-centric methods may not fully model these mappings, as shown by the results in §2.3. Oracle that directly leverages the LVLM's own inference capability consistently outperforms similarity-based methods. Although Oracle's reliance on the ground-truth response makes it impractical for direct inference, it provides an effective way to generate training data for model-centric learning of LVLM reasoning paths. Therefore, we propose the **Task-Aware model for in-COntext Learning (TACO)**, a lightweight, end-to-end framework designed to select ICDs that enhance both local and global task alignment. TACO is trained using data derived from LVLMs and leverages a specialized attention mechanism that models the reasoning patterns guiding task mapping.

Figure 4 illustrates the overall architecture of TACO. Its backbone consists of four transformer decoder blocks. Each triplet example (I_i, Q_i, R_i) from the demonstration library DL is treated as a distinct token. The training dataset D_S is composed of N-shot ICL sequences. During inference, given a query sample and $Inst$, TACO can autoregressively retrieve n samples from DL to configure the optimal n-shot ICL sequence.

Input Embedding. Let x_i denote i-th ICD token (I_i, Q_i, R_i) and $\hat{x}$ denote the query sample $(\hat{I}, \hat{Q})$. In each input sequence, $\hat{x}$ is placed ahead of all x_i. To align with the autoregressive generation process, we use two special tokens, $[BOS]$ and $[EOS]$, to mark the beginning and end of the input sequence during training. These tokens are added to TACO's vocabulary. We also introduce a $[TASK]$ token into the vocabulary and concatenate it with $\hat{x}$ in the input sequence. It acts as a semantic anchor for task mapping. Therefore, for a given S^N, we reconstruct it as a token sequence $([BOS], [TASK]+\hat{x}, x_1, ..., x_N, [EOS]\})$. To filter and balance multimodal features for better mapping construction, we employ a binary fusion module to generate the embedding e_i for x_i:

$$f_i = \sigma(W_f \cdot [E_I(I_i) \oplus E_T(Q_i \oplus R_i)] + b_f), \quad (4)$$

$$e_i = f_i \cdot E_I(I_i) + (1 - f_i) \cdot E_T(Q_i \oplus R_i), \quad (5)$$

where $E_I(\cdot)$ and $E_T(\cdot)$ denote image encoder and text encoder of CLIP. Finally, the input embedding sequence of TACO is presented as follows:

$$e_{S^N} = [e_{\text{BOS}}, \hat{e}, e_1, \ldots, e_N, e_{\text{EOS}}], \quad (6)$$

where e_{BOS} and e_{EOS} are learnable embeddings of $[BOS]$ and $[EOS]$. $\hat{e}$ is a joint representation formed by concatenating the learnable embedding of $[TASK]$ with the embedding of $\hat{x}$ generated using the same fusion module. The index of $\hat{e}$ is always 1, and I_{idx} denotes the index set of e_i.

Task-aware Attention. The task-aware attention in TACO enables dynamic ICL sequence configuration by integrating task mappings into attention computation. Its core is the task guider (TG), an embedding independent of the input sequence, designed to capture fine-grained global task mapping within ICL sequences. TG encodes task intent through initialization by the multimodal fusion of the query sample and instruction:

$$e_{TG}^{(0)} = W_{TG} \cdot (E_I(\hat{I}) \oplus E_T(\hat{Q}) \oplus E_T(Inst')), \quad (7)$$

where $W_{TG} \in \mathbb{R}^{d \times 3d}$ is a learnable weight matrix used to regulate the entire TG. $Inst'$ is a simplified instruction generated by GPT-4o (Appendix D.2).

Task-aware attention is applied selectively to certain layers, denoted as L_{Ta}. At each of these layers, TG steers the attention mechanism by weighting relevance scores, which are derived from the interaction between TG and token embeddings. This
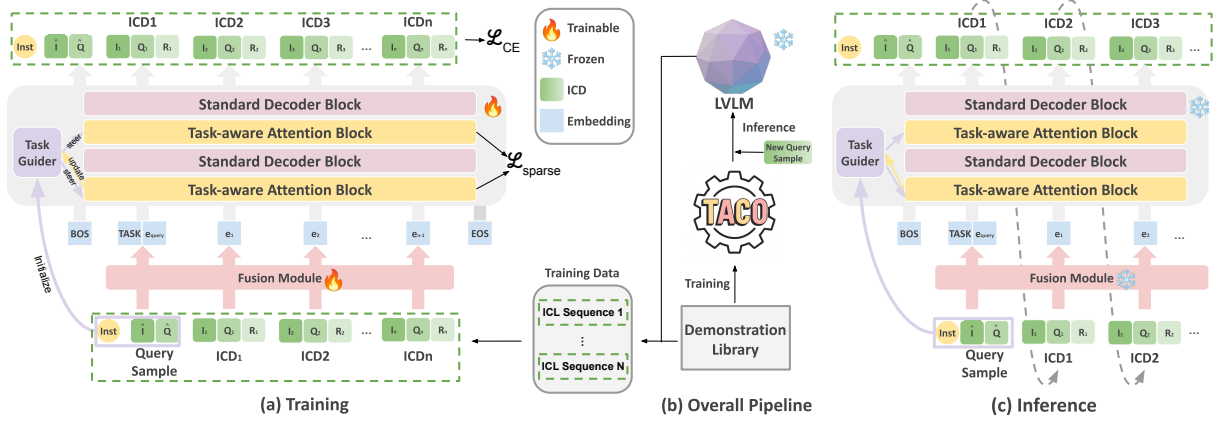
Figure 4: Our overall pipeline, shown in (b), consists of three parts: a demonstration library, TACO, and a pre-trained LVLM. TACO treats each (I, Q, R) example in the demonstration library as a token. (a) shows TACO training using the LVLM-constructed training data. (c) shows that, given a new query sample, TACO autoregressively retrieves samples from the demonstration library to form a high-quality ICL sequence for LVLM inference.

interaction captures the hierarchical relationships between task mappings within the ICL sequence:

$$t_i^{(l)} = \sigma\Big(\text{MLP}^{(l)}\big(e_{TG}^{(l)} \oplus e_i\big)\Big), \qquad (8)$$

where $\text{MLP}^{(l)} \colon \mathbb{R}^{2d} \to \mathbb{R}^d$ is a layer-specific network producing a scalar weight $t_i^l \in [0, 1]$ and σ is the sigmoid function. This weight reflects the degree to which each token contributes to the cohesive task mapping, dynamically adapting TACO's attention to emphasize semantically salient features. It modulates attention logits through a task-aware mask $M^{(l)}$. For intra-ICD tokens, the mask scales pairwise cosine similarities by $-\log(t_i^{(l)})$. For query-ICD tokens, a learnable coefficient α allows $\hat{e}$ to guide attention throughout the sequence. The mask is computed as follows for position (i, j):

$$M_{ij}^{(l)} = \begin{cases} \frac{\text{sim}(e_i, e_j)}{\sqrt{d}} \cdot \big(-\log(t_i^{(l)})\big), & j \le i \text{ and } i, j \in I_{idx}, \\ \frac{\alpha \text{sim}(\hat{e}, e_j)}{\sqrt{d}} \cdot \big(-\log(t_1^{(l)})\big), & i = 1 \text{ and } j \in I_{idx}, \\ -\infty, & \text{otherwise.} \end{cases}$$
$$(9)$$

Here, the first case emphasizes interactions between local task mappings, and the second case enables deep task mapping cohesion. The last case preserves the autoregressive nature. The mask $M^{(l)}$ is integrated into standard attention, forming task-aware attention (TaAttn), as follows:

$$\text{TaAttn}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}} + M^{(l)}\right) V. \qquad (10)$$

In particular, TG is updated between task-aware layers to preserve task mapping, enabling hierarchical refinement from coarse task intent to fine-grained mapping. After processing layer $l \in \mathcal{L}_T$

through residual connections, TG is updated via:

$$e_{TG}^{(l')} = \text{LN}\left(e_{TG}^{(l)} + \text{Attention}(e_{TG}^{(l)}, H^{(l)})\right), \qquad (11)$$

where l' denotes the next task-aware layer in L_{Ta}, $H^{(l)}$ denotes the hidden states of layer l and LN denotes layer normalization. To ensure focused attention patterns, we introduce a sparsity loss that penalizes diffuse distributions:

$$\mathcal{L}_{\text{sparse}} = \sum_{l \in L_{Ta}} \frac{1}{N} \sum_{i=1}^{N} \text{KL}\left(\text{softmax}(M_{i:}^{(l)}) \parallel \mathcal{U}\right), \qquad (12)$$

where $\mathcal{U}$ is a uniform distribution. Minimizing this KL divergence prompts a sharper representation of task-mapping. The total training objective combines the standard cross-entropy loss for sequence generation, sparsity regularization, and L2-norm constraint on TG to prevent overfitting:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_1 \mathcal{L}_{\text{sparse}} + \lambda_2 \|W_{TG}\|_2^2. \qquad (13)$$

Inference and Prompt Construction. After training, TACO can autoregressively select demonstrations from a library and configure ICL sequences. Given a new query sample $\hat{x}$, the input sequence to TACO during inference is $\{[BOS], [TASK] + \hat{x}\}$, where $\hat{x}$ is embedded using the trained fusion module. The shot of the generated sequence, denoted as n, is a user-defined value. It may differ from the shot count N in D_S, as discussed in §5. TACO then selects n ICDs using a beam search strategy with a beam size of 3, producing the optimal n-shot ICL sequence S^n. This sequence is used to

| Methods | VQA | | | Captioning | | Classification | Hybrid | Fast | CLEVR |
	VQAv2 ACC.↑	VizWiz ACC.↑	OK-VQA ACC.↑	Flickr30K CIDEr↑	MSCOCO CIDEr↑	HatefulMemes ROC-AUC↑	ACC.↑	ACC.↑	ACC.↑
RS	60.32	43.38	51.85	93.67	110.81	74.14	17.74	64.04	43.42
I2I	58.29	43.10	51.54	96.02	110.77	76.00	14.69	66.69	41.56
IQ2IQ	61.37	44.96	53.90	94.24	112.04	74.83	34.85	68.26	41.58
IQPR	62.25	44.99	54.58	95.19	113.52	73.62	35.63	67.68	43.98
DEmO	61.10	45.41	55.28	95.21	113.24	74.02	34.56	67.20	42.12
Lever-LM	64.13	48.13	58.33	98.24	118.27	78.86	41.61	67.36	46.51
Ours	**66.75**	**52.07**	**61.54**	**99.62**	**119.47**	**80.59**	**45.22**	**68.73**	**48.45**

Table 1: Results of different ICL sequence configuration methods across 9 datasets, with both training and generated sequences being 4-shot. Each result is the average performance across **five** LVLMs with the same prompt format. The highest scores are highlighted in **bold**. Underlined values indicate the results of the best baselines. Detailed results for each LVLM can be found in Table 8.

construct a prompt for LVLMs, formatted as: $\{Inst; ICD_1, ..., ICD_n; Query\ Sample\}$, which is then used to perform multimodal ICL. Example prompts are provided in Appendix D.3.

4 Experiment

4.1 Training Data Construction and Models

Following standard multimodal ICL evaluation practices (Awadalla et al., 2023), we select six high-quality datasets across three core VL tasks: VQAv2, VizWiz (Gurari et al., 2018), and OK-VQA (Marino et al., 2019) for open-ended VQA; Flickr30K (Young et al., 2014) and MSCOCO (Lin et al., 2014) for captioning; and HatefulMemes for classification. To further assess TACO's abilities in generalized-mapping tasks, we create a mixed-task dataset **Hybrid**, by sampling 5,000 instances from the training set from each above dataset, with validation samples drawn proportionally from their validation sets. We also adopt two challenging image-to-text tasks from the latest multimodal ICL benchmark, VL-ICL (Zong et al., 2024): Fast Open-Ended MiniImageNet[1] (**Fast**) and **CLEVR**.

To construct the high-quality sequence dataset D_S for TACO training from the above datasets, we first reformulate them into (I, Q, R) triplets. Using clustering, we select K samples from their training sets as query samples, forming the query set $\hat{D}$. For each query sample in $\hat{D}$, N ICDs are retrieved from the remaining data using the Oracle method described in §2.3, creating S^N. This retrieval process is further refined through beam search to improve the quality and diversity of D_S. The implementation details are provided in Appendix E.2. All S^N begin with a CoT-style $Inst$,

as detailed in *Beginning1* of Table 4.

Our experiments evaluate four advanced open-source LVLMs: OpenFlamingov2-9B, Idefics2-8B, InternVL2.5-8B (Chen et al., 2024c), and Qwen2.5VL-7B (Bai et al., 2025), as well as a closed-source model, GPT-4V (OpenAI, 2023). Detailed descriptions of the datasets and LVLMs are provided in Appendix E.1.

4.2 Baselines and Implementation Details

We adopt RS and two similarity-based retrieval methods introduced in §2.3 as baselines, along with three previous SOTA configuration methods: IQPR (Li et al., 2024b), DEmO (Guo et al., 2024), and Lever-LM (Yang et al., 2024). Lever-LM is a tiny language model with several standard decoder blocks that performs automatic S^n configuration. As it also requires model training, we treat it as the primary baseline. For a fair comparison, we set Lever-LM's depth to four layers. Details of the baselines are provided in Appendix E.3.

We evaluate ICL sequences on LVLMs using validation sets of the datasets, with the training sequence shot N and the generated sequence shot n set to 4. Query set $\hat{D}$ sizes vary by dataset (Table 5). We utilize the image and text encoders from CLIP-ViT-L/14 to generate image and text embeddings. For all tasks, we employ a unified encoder training strategy: updating only the last three layers while keeping all preceding layers frozen. TACO training employs a cosine annealed warm restart learning scheduler, AdamW optimizer, 1e-4 learning rate, batch size 128, and runs for 20 epochs.

4.3 Main Results

Table 1 summarizes the average performance of ICL in five LVLMs using different methods of configuring the ICL sequence. TACO consistently out-

---
[1]We apply fine-grained adjustments to **Fast** to increase its difficulty; see Appendix E.1 for details.

Configuration	VQA			Captioning		Classification	Hybrid	Fast	CLEVR
	VQAv2	VizWiz	OK-VQA	Flickr30K	MSCOCO	HatefulMemes			
Full TACO	**66.75**	**52.07**	**61.54**	**99.62**	**119.47**	**80.59**	**45.22**	**68.73**	**48.45**
(a) w/o [TASK] token	64.58	50.24	60.26	98.39	118.04	79.51	42.83	67.18	46.38
(b) w/o TG updates	63.53	48.36	59.01	97.65	117.84	77.24	40.56	65.27	44.79
(c) w/o $\mathcal{L}_{sparse}$	63.79	51.25	60.95	98.19	118.10	78.29	42.33	65.93	45.81
(d) w/o $\|W_{TG}\|_2^2$	62.71	48.29	58.41	98.45	117.72	76.35	38.40	66.26	44.28
(e) Random initialization	59.46	42.97	54.59	94.67	111.52	74.48	34.29	60.38	42.52
(f) w/o $\hat{I}$	64.10	49.61	59.65	97.19	115.28	78.50	41.54	66.45	45.08
(g) w/o $\hat{Q}$	62.54	47.24	59.47	96.95	114.73	76.83	40.22	66.07	44.93
(h) w/o $Inst'$	62.68	48.02	60.08	98.32	117.90	77.32	40.75	66.73	45.16

Table 2: Results of the ablation study on task mapping augmentation of TACO. Specifically, (a)-(d) correspond to diverse task-aware attention construction, (e)-(h) to diverse TG initialization.
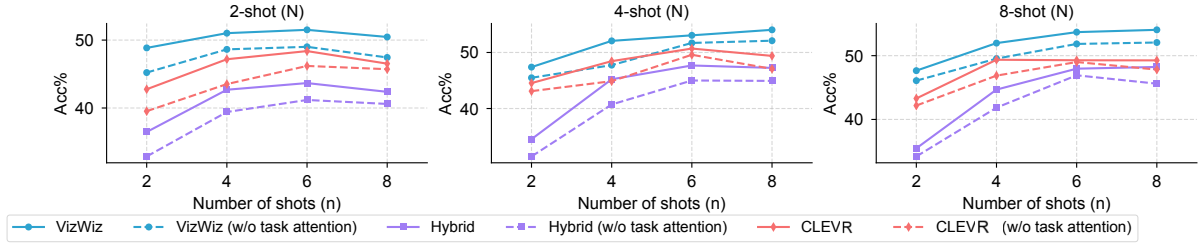

Figure 5: Results of TACO with and without task-aware attention under different N-n settings across three datasets, where N is the training sequence shot and n is the generation sequence shot.

performs all baselines across all nine datasets, highlighting its robustness and effectiveness in fully leveraging the potential of LVLMs for diverse multimodal ICL scenarios. Notably, TACO delivers particularly strong results in generalized-mapping tasks, achieving an average improvement of 3.26% in VQA tasks, with the second highest gain of 3.61% observed on **Hybrid**. These results demonstrate that strengthening task mapping enhances the autoregressive generation process of language models, equipping them with a broader understanding and enabling the construction of more precise cohesive task mappings. In Appendix E.4, we present more evaluations of how ICL sequence configuration affects LVLM using per-model data and include efficiency analyses of TACO to show its low computational overhead.

5 Ablation Study and Discussions

In this section, we examine the impact of task-aware attention and reveal, from a task-mapping perspective, how it enhances ICL performance.

Table 2 shows that each ablated component induces a complete performance degradation. TG, initialized by fusing the query's bimodal context with instruction semantics, establishes a task intent that aligns with the observation of §2: global mapping synthesis relies on query-driven grounding.

Jointly anchored by the $[TASK]$ token, this intent prevents local mapping drift during autoregressive generation but also enables dynamic refinement through layered attention updates. By iteratively resolving coarse task boundaries into fine-grained patterns, TG harmonizes intra-sequence dependencies and query-context interactions, forming a feedback loop where each retrieved ICD sharpens global mapping cohesion. In conclusion, task-aware attention effectively encodes task mapping as a dynamic attention-driven process, transcending static ICD aggregation to achieve consistent performance improvements in multimodal ICL.

To gain a deeper understanding of the role of task mapping throughout from training to inference, we explore different combinations of training and generation shots. Our findings are as follows:

Task mapping consistently enhances multimodal ICL. Figure 5 shows that across all N-n combinations, task-aware attention always improves performance, highlighting the value of focusing ICL sequences on task mapping.

Cohesion remains robust as shots increase. For specific-mapping tasks (e.g., **CLEVR**), when N is fixed, performance gains diminish as n increases, while generalized-mapping tasks generally maintain steady improvements. This arises from each new ICD's unique contribution to the global

Method	VQAv2	VizWiz	OK-VQA	Flickr30K	MSCOCO	HatefulMemes	Hybrid
TACO (RS)	62.38	47.69	54.47	98.31	115.83	76.49	38.62
TACO (I2I)	61.95	47.28	53.86	98.74	117.25	77.46	36.90
TACO (IQ2IQ)	64.37	50.18	59.23	99.17	118.68	78.93	41.05
TACO (Oracle)	**66.75**	**52.07**	**61.54**	**99.62**	**119.47**	**80.59**	**45.22**

Table 3: Results of TACO under diverse training sequence construction strategies.

task mapping, potentially deepening it rather than yielding diminishing returns on a specific mapping.

Task mapping enables flexibility in N and n. Although task-aware attention works best when N equals n, the cohesive design of task mapping allows TACO to effectively interpolate and extrapolate sequence shots across a flexible range of values. This adaptability ensures performance across diverse training data and enhances the model's potential for practical multimodal ICL applications, where flexibility and scalability are critical.

Next, we investigate the construction of training sequences for TACO. In the main experiments, we use Oracle that approximates an LVLM's optimal multimodal ICL. Training TACO on sequences generated by Oracle enables it to learn how to configure ICL inputs with correct task mappings, especially at the global level across the demonstration set. To test the robustness of this learning process, we replace Oracle with similarity-based methods and examine how this change affects TACO's accuracy in inferring task mappings.

As TACO is training-based, one of the most important aspects is high-quality data. Table 3 demonstrates that using Oracle as the method to construct the training data is optimal. Since our approach leverages TACO to capture how LVLMs understand and utilize task mapping, the training data that best reflects the internal mechanisms of the LVLM is most effective. Moreover, it can be observed that TACO, when combined with RS, I2I, and IQ2IQ, brings significant performance improvements over using RS, I2I, and IQ2IQ in isolation. This indicates that our method can mitigate the inherent limitations of retrieval strategies through training, further enhancing the practical value of TACO.

In Appendix F, we conduct additional ablation studies on the construction of input embeddings as well as the format and position of $Inst$. We also evaluate TACO's extension to NLP and text-to-image tasks. Furthermore, we revisit the theoretical framework introduced in §2 through TACO,

further confirming its robustness. Together, these experiments demonstrate that the improvements that TACO brings to multimodal ICL are derived from its task-mapping-guided configuration.

6 Related Works

Interpreting ICL. The mechanisms of ICL are crucial to better employing it (Gao et al., 2021; Dong et al., 2024; Li, 2025). Min et al. (2022) attribute ICL's success to explicit information in ICDs like label space and input distribution, while Zhou et al. (2023) emphasize the importance of input-output mappings. To find a unified solution, Wei et al. (2023) and Pan et al. (2023) disentangle ICL into Task Recognition and Task Learning. Zhao et al. (2024) further propose a two-dimensional coordinate system to explain ICL behavior via two orthogonal variables: similarity in ICDs and LLMs' ability to recognize tasks. However, these studies are often confined to specific-mapping tasks with small label spaces and struggle to address complex multimodal scenarios. We present related work on configuring ICL sequences in Appendix A.

7 Conclusion

In this work, we systematically demonstrate the principles and critical role of task mapping within ICL sequences for enabling effective multimodal ICL in LVLMs. These insights further motivate the use of task mapping to explore more effective ICL sequence configuration strategies that truly align model learning behavior and internal demands. To this end, we propose a transformer-based model, TACO, which employs task-aware attention to deeply integrate task mapping into the autoregressive process, thereby optimizing sequence configuration. Experiments show consistent outperformance over SOTA baselines, particularly in generalized-mapping tasks. This study not only presents a practical model but also provides the multimodal ICL community with a new and reliable research direction.

Limitations

Perspectives from cognitive science are not yet incorporated. In this work we propose task mapping, which represents an abstract inference process performed by an LVLM in its latent space. This concept aligns with themes in cognitive science. By thoroughly examining task mapping we may discover ways to equip LVLMs with more advanced cognitive capabilities. However, our current study does not incorporate cognitive science theory or pursue interdisciplinary exploration, which somewhat limits the impact of task mapping. In future research we will explore cross disciplinary integration based on task mapping.

Our analysis does not examine the internal mechanisms through which task mapping is realized. This study does not delve into the role of LVLMs' internal attention mechanisms and hidden state in capturing and utilizing task mapping. Investigating how task mapping manifests within attention layers could uncover deeper connections between sequence configuration and model reasoning, offering another promising avenue for our future work.

Acknowledgements

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Ruichuan An, Sihan Yang, Ming Lu, Renrui Zhang, Kai Zeng, Yulin Luo, Jiajun Cao, Hao Liang, Ying Chen, Qi She, et al. 2024. Mc-llava: Multi-concept personalized vision-language model. *arXiv preprint arXiv:2411.11706*.

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. 2023. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Rahul Atul Bhope, Praveen Venkateswaran, KR Jayaram, Vatche Isahagian, Vinod Muthusamy, and Nalini Venkatasubramanian. 2025. Optiseq: Ordering examples on-the-fly for in-context learning. *arXiv preprint arXiv:2501.15030*.

Haoran Chen, Junyan Lin, Xinhao Chen, Yue Fan, Xin Jin, Hui Su, Jianfeng Dong, Jinlan Fu, and Xiaoyu Shen. 2025. Rethinking visual layer selection in multimodal llms. *arXiv preprint arXiv:2504.21447*.

Wentong Chen, Yankai Lin, ZhenHao Zhou, HongYun Huang, Yantao Jia, Zhao Cao, and Ji-Rong Wen. 2024a. Icleval: evaluating in-context learning ability of large language models. *arXiv preprint arXiv:2406.14955*.

Yunmo Chen, Tongfei Chen, Harsh Jhamtani, Patrick Xia, Richard Shin, Jason Eisner, and Benjamin Van Durme. 2024b. Learning to retrieve iteratively for in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7156–7168.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. 2024c. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Caoyun Fan, Jidong Tian, Yitian Li, Hao He, and Yaohui Jin. 2024. Comparable demonstrations are important in in-context learning: A novel perspective on demonstration selection. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10436–10440.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Shivam Garg, Dimitris Tsipras, Percy Liang, and Gregory Valiant. 2023. What can transformers learn in-context? a case study of simple function classes. *arXiv preprint arXiv:2208.01066*.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6904–6913.

Tiancheng Gu, Kaicheng Yang, Ziyong Feng, Xingjun Wang, Yanzhao Zhang, Dingkun Long, Yingda Chen, Weidong Cai, and Jiankang Deng. 2025. Breaking the modality barrier: Universal embedding learning with multimodal llms. *arXiv preprint arXiv:2504.17432*.

Qi Guo, Leiyu Wang, Yidong Wang, Wei Ye, and Shikun Zhang. 2024. What makes a good order of examples in in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14892–14904, Bangkok, Thailand. Association for Computational Linguistics.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. 2018. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Dan Iter, Reid Pryzant, Ruochen Xu, Shuohang Wang, Yang Liu, Yichong Xu, and Chenguang Zhu. 2023. In-context demonstration selection with cross entropy difference. *arXiv preprint arXiv:2305.14726*.

Hong Jun Jeon, Jason D Lee, Qi Lei, and Benjamin Van Roy. 2024. An information-theoretic analysis of in-context learning. *arXiv preprint arXiv:2401.15530*.

Jiale Kang, Ziyin Yue, Qingyu Yin, Jiang Rui, Weile Li, Zening Lu, and Zhouran Ji. 2025. Modrwkv: Transformer multimodality in linear time. *arXiv preprint arXiv:2505.14505*.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. 2020. The hateful memes challenge: Detecting hate speech in multimodal memes. *Advances in neural information processing systems*, 33:2611–2624.

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. 2024. What matters when building vision-language models? *Advances in Neural Information Processing Systems*, 37:87874–87907.

Jian Li, Weiheng Lu, Hao Fei, Meng Luo, Ming Dai, Min Xia, Yizhang Jin, Zhenye Gan, Ding Qi, Chaoyou Fu, et al. 2024a. A survey on benchmarks of multimodal large language models. *arXiv preprint arXiv:2408.08632*.

Jiaqian Li, Qisheng Hu, Jing Li, and Wenya Wang. 2025a. Stare at the structure: Steering icl exemplar selection with structural alignment. *arXiv preprint arXiv:2508.20944*.

Li Li, Jiawei Peng, Huiyi Chen, Chongyang Gao, and Xu Yang. 2024b. How to configure good in-context sequence for visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26710–26720.

Wei Li, Renshan Zhang, Rui Shao, Jie He, and Liqiang Nie. 2025b. Cogvla: Cognition-aligned vision-language-action model via instruction-driven routing & sparsification. *arXiv preprint arXiv:2508.21046*.

Yanshu Li. 2025. Advancing multimodal in-context learning in large vision-language models with task-aware demonstrations. *arXiv preprint arXiv:2503.04839*.

Yanshu Li, Yi Cao, Hongyang He, Qisen Cheng, Xiang Fu, Xi Xiao, Tianyang Wang, and Ruixiang Tang. 2025c. M^{2}iv: Towards efficient and fine-grained multimodal in-context learning via representation engineering. In *Second Conference on Language Modeling*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision– ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*.

Man Luo, Xin Xu, Yue Liu, Panupong Pasupat, and Mehran Kazemi. 2024a. In-context learning with retrieved demonstrations for language models: a survey. *arXiv preprint arXiv:2401.11624*.

Meng Luo, Hao Fei, Bobo Li, Shengqiong Wu, Qian Liu, Soujanya Poria, Erik Cambria, Mong-Li Lee, and Wynne Hsu. 2024b. Panosent: A panoptic sextuple extraction benchmark for multimodal conversational aspect-based sentiment analysis. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7667–7676.

Xinxi Lyu, Sewon Min, Iz Beltagy, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Z-icl: zero-shot in-context learning with pseudo-demonstrations. *arXiv preprint arXiv:2212.09865*.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204.

Reid McIlroy-Young, Katrina Brown, Conlan Olson, Linjun Zhang, and Cynthia Dwork. 2024. Order-independence without fine tuning. *Advances in Neural Information Processing Systems*, 37:72818–72839.

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? *arXiv preprint arXiv:2202.12837*.

Shiwen Ni, Dingwei Chen, Chengming Li, Xiping Hu, Ruifeng Xu, and Min Yang. 2023. Forgetting before learning: Utilizing parametric arithmetic for knowledge updating in large language models. *arXiv preprint arXiv:2311.08011*.

nostalgebraist. 2020. Interpreting gpt: the logit lens. *LessWrong*.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. 2022. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jane Pan, Tianyu Gao, Howard Chen, and Danqi Chen. 2023. What in-context learning "learns" in-context: Disentangling task recognition and task learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8298–8319, Toronto, Canada. Association for Computational Linguistics.

Zhenyu Pan, Yutong Zhang, Jianshu Zhang, Haoran Lu, Haozheng Luo, Yuwei Han, Philip S Yu, Manling Li, and Han Liu. 2025. Fairreason: Balancing reasoning and social bias in mllms. *arXiv preprint arXiv:2507.23067*.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-okvqa: A benchmark for visual question answering using world knowledge. In *European conference on computer vision*, pages 146–162. Springer.

Sarah Schwettmann, Neil Chowdhury, Samuel Klein, David Bau, and Antonio Torralba. 2023. Multimodal neurons in pretrained text-only transformers. In *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 2854–2859.

Quan Sun, Qiying Yu, Yufeng Cui, Fan Zhang, Xiaosong Zhang, Yueze Wang, Hongcheng Gao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. 2024. Emu: generative pretraining in multimodality. *arXiv preprint arXiv:2307.05222*.

Minh-Hao Van, Xintao Wu, et al. 2024. In-context learning demonstration selection via influence analysis. *arXiv preprint arXiv:2402.11750*.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Liang Wang, Nan Yang, and Furu Wei. 2024. Learning to retrieve in-context examples for large language models. *arXiv preprint arXiv:2307.07164*.

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

Yang Wu, Shilong Wang, Hao Yang, Tian Zheng, Hongbo Zhang, Yanyan Zhao, and Bing Qin. 2023a. An early evaluation of gpt-4v(ision). *arXiv preprint arXiv:2310.16534*.

Zhiyong Wu, Yaoxiang Wang, Jiacheng Ye, and Lingpeng Kong. 2023b. Self-adaptive in-context learning: an information compression perspective for in-context example selection and ordering. *arXiv preprint arXiv:2212.10375*.

Xu Yang, Yingzhe Peng, Haoxuan Ma, Shuo Xu, Chi Zhang, Yucheng Han, and Hanwang Zhang. 2024. Lever lm: configuring in-context sequence to lever large vision language models. *arXiv preprint arXiv:2312.10104*.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78.

Yu Yuan, Lili Zhao, Kai Zhang, Guangting Zheng, and Qi Liu. 2024. Do llms overcome shortcut learning? an evaluation of shortcut challenges in large language models. *arXiv preprint arXiv:2410.13343*.

Anhao Zhao, Fanghua Ye, Jinlan Fu, and Xiaoyu Shen. 2024. Unveiling in-context learning: a coordinate system to understand its working mechanism. *arXiv preprint arXiv:2407.17011*.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

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

Yongshuo Zong, Ondrej Bohdal, and Timothy Hospedales. 2024. Vl-icl bench: the devil in the details of multimodal in-context learning. *arXiv preprint arXiv:2403.13164*.

A Additional Related Works

Configuring ICL sequences. To configure high-quality ICL sequences that bolster multimodal ICL in LVLMs, researchers have explored numerous methods, with metric-centric approaches emerging as the most prominent (McIlroy-Young et al., 2024). The most direct metric for both implementation and evaluation is similarity. In this category, methods select ICDs from a demonstration library by comparing their embeddings, an approach widely used in retrieval augmented generation (RAG) systems (Luo et al., 2024a; Chen et al., 2024b). Retrieval strategies based on semantic entropy have also been applied to tasks demanding more fine-grained selection criteria (Wu et al., 2023b; Jeon et al., 2024). To accommodate more complex tasks, several novel metrics have been introduced. Guo et al. (2024) define an influence score, which quantifies the change in model confidence induced by each demonstration, and combine this score with entropy to configure ICL sequences. Bhope et al. (2025) leverages log probabilities of LLM-generated outputs to systematically prune the search space of possible orderings. Although these human-designed metrics can partially capture each ICD's contribution to latent reasoning, the black-box nature of LVLMs prevents them from fully reflecting the model's internal inference. For example, similarity-based methods may not provide LLMs with deep task mappings (Liu et al., 2021; Li et al., 2024b). Approaches designed to mitigate ICD bias can also inadvertently introduce new biases (Lyu et al., 2023; Yuan et al., 2024). Model-centric methods have also emerged later, employing multiple models for more demanding selection (Wu et al., 2023b; Wang et al., 2024; Van et al., 2024). These methods are not end-to-end and overly focus on ICD selection over ordering. One work closely connected to ours is Yang et al. (2024), which introduces a tiny language model composed of two encoder blocks to automatically select and order ICDs. However, its effectiveness on complex tasks is constrained by a lack of deep insight into task mapping.

B Formal Theoretical Definition

In §2.1, we provide simple definitions of local and global task mappings for clarity. Here, we develop a more complete theoretical analysis of task mapping.

Let $\mathcal{I}$ denote the image space, $\mathcal{Q}$ the query space, and $\mathcal{R}$ the response space. Define the space of deterministic mappings

$$\mathcal{F} = \big\{ f : \mathcal{I} \times \mathcal{Q} \to \mathcal{R} \big\}.$$

The pretrained LVLM M_θ induces a conditional distribution over $\mathcal{R}$ given an n-shot ICL prompt:

$$p_\theta(r \mid S^n) = \mathrm{Pr}_{M_\theta}\big(r \mid \mathrm{Inst}; D_1, \ldots, D_n; (\hat{I}, \hat{Q})\big), \tag{14}$$

where $S^n = (\mathrm{Inst}; D_1, \ldots, D_n; (\hat{I}, \hat{Q}))$ and $D_i = (I_i, Q_i, R_i)$.

Definition B.1 (Local Task Mapping). Each demonstration D_i induces a local mapping $f_i \in \mathcal{F}$ defined by

$$f_i = \arg\max_{f \in \mathcal{F}} \mathbb{E}_{(I,Q,r) \sim D_i}\big[\mathbb{1}\{f(I,Q) = r\}\big] \implies f_i(I_i, Q_i) = R_i, \tag{15}$$

which under M_θ equivalently satisfies

$$f_i(I, Q) = \arg\max_{r \in \mathcal{R}} p_\theta\big(r \mid \mathrm{Inst}; D_{<i}, (I, Q)\big). \tag{16}$$

Definition B.2 (Global Task Mapping). The global mapping $\hat{f} \in \mathcal{F}$ induced by the full sequence S^n is

$$\hat{f} = \arg\max_{f \in \mathcal{F}} \mathbb{E}_{r \sim p_\theta(\cdot \mid S^n)}\big[\mathbb{1}\{f(\hat{I}, \hat{Q}) = r\}\big] \implies \hat{f}(\hat{I}, \hat{Q}) = \hat{R}, \tag{17}$$

which reduces to

$$\hat{f}(\hat{I}, \hat{Q}) = \arg\max_{r \in \mathcal{R}} p_\theta\big(r \mid S^n\big). \tag{18}$$

Definition B.3 (Mapping Composition). Introduce the composition operator $C_\theta : \mathcal{F}^n \times \{\mathrm{Inst}\} \to \mathcal{F}$ such that

$$\hat{f} = C_\theta\big(f_1, \ldots, f_n; \mathrm{Inst}\big), \tag{19}$$

capturing how the model integrates local mappings into the global mapping.

Definition B.4 (Specific vs. Generalized Mapping).

Specific-mapping: $f_1 = f_2 = \cdots = f_n,$

Generalized-mapping: $\exists i \neq j, \; f_i \neq f_j.$

C Vision-language In-context Learning

C.1 Demonstration Configuring Details

(a) **Open-ended VQA**: The query Q_i is the single question associated with the image I_i, while the response Ri is the answer to the question, provided
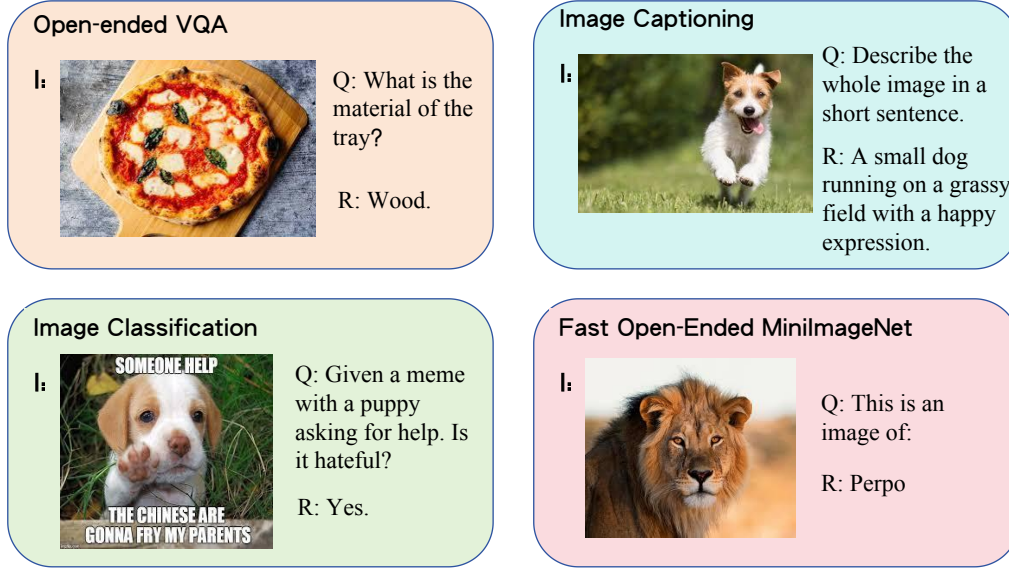
Figure 6: The visualization of (I, Q, R) triplets for Open-ended VQA, image captioning, image classification and Fast Open-ended MiniImageNet.

as a short response. For the query sample, $\hat{Q}$ represents the question related to the image $\hat{I}$, and $\hat{R}$ is the expected output of the model.

(b) **Image Captioning**: Both Q_i and $\hat{Q}$ are set as short prompts instructing the LVLM to generate a caption for the given image, such as "Describe the whole image in a short sentence. " The response R_i corresponds to the actual caption of the image.

(c) **Image Classification**: Both Q_i and $\hat{Q}$ provide the textual information paired with the image, followed by a directive requiring the model to classify based on the provided image-text pairs. The response R_i is the predefined class label.

(d) **Fast Open-ended MiniImageNet**: Both Q_i and $\hat{Q}$ are set as short prompts instructing the LVLM to recognize the object in image, such as "This is an image of:" The response R_i is the self-defined label.

(e) **CLEVR Counting Induction**: Both Q_i and $\hat{Q}$ are implicit texts in the form of "attribute: value" pairs. The response R_i is the number of objects matching the pairs.

For all the tasks mentioned above, since the ground-truth answers are not visible to the LVLM during reasoning, all $\hat{R}$ are set to blank. The visualization of (I, Q, R) triplets for the four tasks is shown in Figure 6.

C.2 In-context Lens

To visualize how LVLMs' internal token outputs evolve during ICL on specific-mapping tasks, we introduce the in-context lens, an adaptation of the logit lens (nostalgebraist, 2020). Like its predecessor, in-context lens projects each layer's final token embedding back into the text vocabulary. Because local mappings in specific-mapping tasks are largely uniform, we can vary task difficulty to distinguish shallow recognition from deep recognition. In the HatefulMemes example, shallow recognition corresponds to the binary classification decision, while deep recognition requires detecting harmful content. We therefore define four anchor categories by selecting representative keywords: "Shallow" represents superficial task understanding, focusing on general or surface-level concepts. Anchor words include "category," "judge," "label," "identify," and "predict." "Deep" indicates a more profound comprehension of the task, capturing nuanced or context-sensitive meanings. Anchor words include "hateful," "offensive," "biased," "harmful," and "inappropriate." "Correct" corresponds to the correct answer for the query sample. "Wrong" represents the incorrect answer, opposite to "Correct." We then compute, for each layer, the relative probability of the top three relevant decoded tokens falling into each category (summing to 100%) and visualize the results as pie charts. Figure 7 shows these charts corresponding to the visualizations in Figure 1(c).
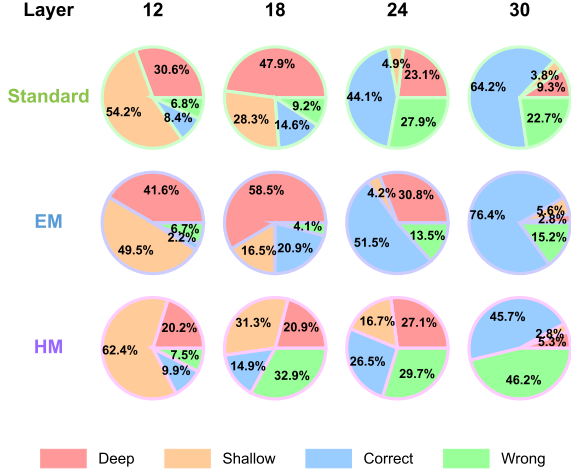
749

Figure 7: Visualization of the evolving internal representations of an LVLM on the HatefulMemes dataset during ICL, analyzed through the in-context lens.

C.3 Oracle

Oracle uses the same LVLM $\mathcal{M}$ for both configuring the ICL sequences and performing ICL. This method aims to construct high-quality ICL sequences by iteratively evaluating and selecting demonstrations based on their contribution to the model's predictive performance. Given the ground-truth response $\hat{R} = (\hat{R}^{(1)}, ..., \hat{R}^{(t)})$ of the query sample, Oracle computes the log-likelihood score $\mathcal{C}_{\mathcal{M}}(S^n)$ for a sequence S^n with n ICDs, defined as:

$$\mathcal{C}_{\mathcal{M}}(S^n) = \sum_t log P_{\mathcal{M}}(\hat{R}^{(t)} \mid S^n, \hat{R}^{(1:t-1)}), \tag{20}$$

where $\mathcal{M}$ denotes the LVLM. This score measures how effectively the model predicts the ground-truth response $\hat{R}$ given the current ICL sequence S^n.

The configuration process begins with an empty sequence S^0 and iteratively selects demonstrations. At each step n, a demonstration x_n is chosen from the library D to maximize the incremental gain in the log-likelihood score:

$$x_n = \underset{x \in D}{argmax}[\mathcal{C}_{\mathcal{M}}(S^{n-1} + x) - \mathcal{C}_{\mathcal{M}}(S^{n-1})]. \tag{21}$$

This greedy optimization process ensures that each selected demonstration contributes optimally to the sequence. Unlike simple similarity-based methods, Oracle evaluates the overall impact of each candidate demonstration on the sequence's quality.

C.4 Ablation Settings

To systematically evaluate the impact of task mapping in multimodal in-context learning (ICL), we design controlled ablation settings that selectively perturb key factors such as **label reliability** and **visual modality**. Below, we provide detailed descriptions of each setting's implementation.

1. **Label Reliability**
   - *Wrong Labels (WL)*: To evaluate the reliance on explicit label correctness, we invert 75% R_i labels (yes↔no) in the ICL sequence. This setting disrupts direct label-based learning while maintaining the overall task structure, allowing us to examine whether LVLMs primarily depend on task mapping rather than correct labels.

2. **Visual modality**
   - *Blur Images (BI)*: To investigate the role of visual information clarity, we apply Gaussian blur to the images I_i in the ICL sequence. This degrades fine-grained details while preserving overall structure, allowing us to examine the impact of visual degradation on task mapping.
   - *BI on Query Image (BI(Î))*: Instead of applying blur to the entire ICL sequence, (BI)$\hat{I}$ applies Gaussian blur only to the query image $\hat{I}$. This setting helps isolate the effect of degraded query information on task mapping performance.

3. **Query Enhancement**
   - *Easier Mapping on Query (EM($\hat{Q}$))*: This setting enhances the query text $\hat{Q}$ by incorporating explicit task guidance to facilitate task mapping. Instead of modifying the ICL sequence, EM($\hat{Q}$) provides additional textual hints that reinforce task semantics, allowing us to measure whether improved query understanding compensates for suboptimal ICD configurations.

C.5 Task Mapping Cohesion Metrics

C.5.1 Disruption Gap (Δ)

To measure the impact of individual ICDs on sequence-level performance and assess task mapping cohesion, we define the Disruption Gap (Δ) as the magnitude of performance change caused by replacing a single ICD in the sequence.

For each ICD $x_i = (I_i, Q_i, R_i)$ in the sequence S^n, a replacement ICD $x_j = (I_j, Q_j, R_j)$ is selected from the same dataset based on the highest joint similarity of their image and query embeddings (IQ2IQ). The modified sequence $S_{\text{replaced},i}$ is

then constructed by replacing x_i with x_j.

The Disruption Gap for the i-th ICD is defined as the absolute difference in performance before and after the replacement:

$$\Delta_i = \left| \mathcal{L}(S) - \mathcal{L}(S_{\text{replaced},i}) \right|, \qquad (22)$$

where $\mathcal{L}(\cdot)$ represents the performance metric of the sequence (e.g., accuracy).

For a sequence S with N ICDs, the overall Disruption Gap is computed as the average Δ_i across all N ICDs:

$$\Delta = \frac{1}{N} \sum_{i=1}^{N} \Delta_i. \qquad (23)$$

To ensure the robustness of Δ and to account for potential variability in replacement effects, we conduct repeated experiments. This metric quantifies the sequence's cohesion by assessing the sensitivity of the overall performance to individual replacements. A higher Δ indicates that the sequence has stronger cohesion, as replacing an ICD results in larger performance changes.

C.5.2 Order Sensitivity (σ)

For an ICL sequence S^n, we generate K independent random permutation of it:

$$S^n_{\text{permute},1}, S^n_{\text{permute},2}, \ldots, S^n_{\text{permute},K}, \quad K = 10. \qquad (24)$$

Then we compute the accuracy for each permuted sequence $k = 1, 2, \ldots, K$:

$$\text{Acc}\left(S^n_{\text{permute},k}\right) = \frac{\text{Correct Predictions}}{\text{Total Predictions}}. \qquad (25)$$

Then calculate the mean accuracy across all permutations:

$$\mu = \frac{1}{K} \sum_{k=1}^{K} \text{Acc}(S^n_{\text{permute},k}). \qquad (26)$$

Finally, compute the standard deviation of accuracies as σ:

$$\sigma = \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left(\text{Acc}(S^n_{\text{permute},k}) - \mu \right)^2}. \qquad (27)$$

C.5.3 Metric Analysis

Δ and σ together constitute a rigorous framework for quantifying the cohesion of task mappings in in-context learning. The Disruption Gap is defined as the mean absolute degradation in task performance when each ICD is replaced by its nearest neighbor in the learned representation space; this metric directly captures the indispensability of individual local mappings for preserving the semantics of the overall task. A larger value of Δ implies that each ICD contributes uniquely and cannot be substituted without harming global inference. Order Sensitivity is computed as the standard deviation of task accuracy across multiple random permutations of ICD order; this metric assesses the invariance of the global mapping to the structural arrangement of examples. A smaller value of σ indicates that the inferred mapping remains stable regardless of ICL sequence, reflecting intrinsic consistency among local mappings. By combining Δ and σ, one obtains a complementary view in which Δ measures discriminative necessity and σ measures structural resilience, thus ensuring that a truly cohesive task mapping exhibits both strong local-to-global alignment and robustness to variations in ICD composition.

C.6 Case Study

In Figure 8, we present four examples representing the four typical types of ICL sequences in generalized-mapping tasks.

D Method

D.1 CLIP Encoders

CLIP employs two distinct encoders: one for images and another for text. The image encoder transforms high-dimensional visual data into a compact, low-dimensional embedding space, using architectures such as a ViT. Meanwhile, the text encoder, built upon a Transformer architecture, generates rich textual representations from natural language inputs.

CLIP is trained to align the embedding spaces of images and text through a contrastive learning objective. Specifically, the model optimizes a contrastive loss that increases the cosine similarity for matched image-text pairs, while reducing it for unmatched pairs within each training batch. To ensure the learning of diverse and transferable visual concepts, the CLIP team curated an extensive dataset comprising 400 million image-text pairs, allowing the model to generalize effectively across various downstream tasks.

In our experiments, we employ the same model, CLIP-ViT-L/14, using its image and text encoders
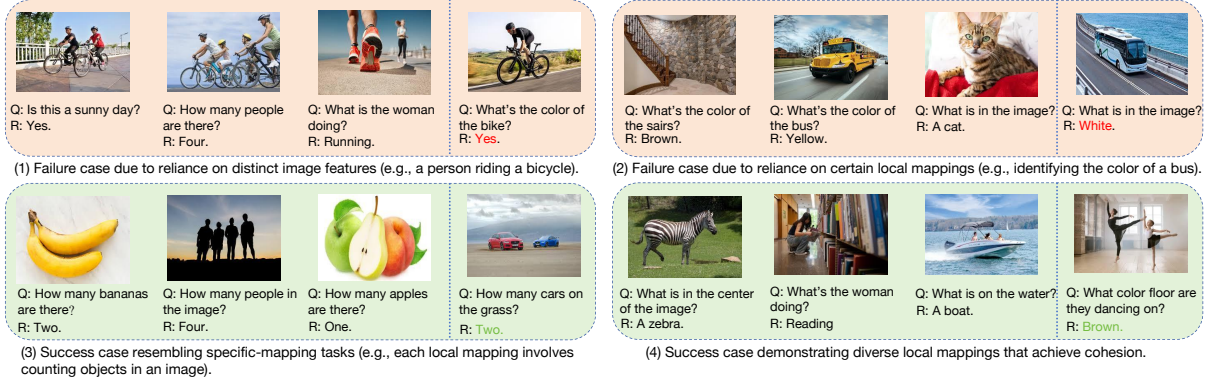
Figure 8: Four types of ICL sequences in the generalized-mapping tasks.

to generate the image and text embeddings for each demonstration, ensuring consistency in cross-modal representations. The model employs a ViT-L/14 Transformer architecture as the image encoder and a masked self-attention Transformer as the text encoder. We experimented with several strategies for training the CLIP encoder and found that training only the last three layers of the encoder offers the best cost-effectiveness.

D.2 Instruction

The $Inst$ generated by GPT-4o in the main experiment is "You will be provided with a series of image-text pairs as examples and a question. Your task involves two phases: first, analyze the provided image-text pairs to grasp their context and try to deeply think about what the target task is; second, use this understanding, along with a new image and your knowledge, to accurately answer the given question." This content demonstrates great orderliness and can act as a good general semantic guide for ICDs and the query sample. This style is named chain-of-thought (CoT) (Li et al., 2025a).

To incorporate the semantic information of $Inst$ and strengthen task representation during the ICL sequence configuration process, we use GPT-4o to generate simplified versions of these $Inst$ and integrate their embeddings into the task guider, which are indicated by $Inst'$. The prompt we use is as follows: *"This is an instruction to enable LVLMs to understand and perform a multimodal in-context learning task. Please simplify it by shortening the sentence while preserving its function, core meaning, and structure. The final version should be in its simplest form, where removing any word would change its core meaning."* This simplification process allows us to investigate how the semantic infor-

mation density in the instruction impacts TACO's sequence configuration ability and the performance of LVLMs in ICL. The results show that simplifying the instruction in a prompt before embedding it in the task guider significantly improves the quality of sequence generation. It also helps to avoid issues caused by too long instructions.

As shown in Table 4, we use GPT-4o to rewrite $Inst$, placing it at the middle and the end of a prompt, altering its semantic structure accordingly while keeping its CoT nature. The table also presents two other tested styles of instructions placed at the beginning of the prompt: Parallel Pattern Integration (PPI) and System-Directive (SD). PPI emphasizes simultaneous processing of pattern recognition and knowledge integration, focusing on dynamic pattern repository construction rather than sequential reasoning. SD structures input as a formal system protocol with defined parameters and execution flows, prioritizing systematic processing over step-by-step analysis. These two forms have also been proven to be effective in previous ICL work. We use them to study the robustness of TACO and various LVLMs to different instruction formats.

D.3 Prompt Details

The prompts constructed based on S^n all follow the format:

$$(Inst; ICD_1, ..., ICD_n; QuerySample).$$

Each ICD's query begins with "Query:" and its response starts with "Response:". The query sample concludes with "Response:", prompting the LVLM to generate a response. Depending on the input format required by different LVLMs, we may also

Inst	Details
Beginning1 (CoT)	You will be provided with a series of image-text pairs as examples and a question. Your task involves two phases: first, analyze the provided image-text pairs to grasp their context and try to deeply think about what the target task is; second, use this understanding, along with a new image and your knowledge, to accurately generate the response to the given query.
Beginning2 (PPI)	Construct a dynamic pattern repository from image-text samples, then leverage this framework alongside your knowledge base for concurrent visual analysis and query resolution. The key is parallel processing - your pattern matching and knowledge integration should happen simultaneously rather than sequentially.
Beginning3 (SD)	SYSTEM DIRECTIVE Input Stream: Example Pairs → New Image + Query Process: Pattern Extract → Knowledge Merge → Visual Analysis → Response Critical: All exemplar patterns must inform final analysis Priority: Context preservation essential
Middle (CoT)	Now you have seen several examples of image-text pairs. Next, you will be given a question. Your task involves two phases: first, revisit the above image-text pairs and try to deeply think about what the target task is; second, use this understanding, along with a new image and your knowledge, to accurately generate the response to the given question.
End (CoT)	Now you have seen several examples of image-text pairs and a question accompanied by a new image. Your task involves two phases: first, revisit the provided examples and try to deeply think about what the target task is; second, use this understanding, the new image, and your knowledge to accurately generate the response of the given question.
Beginning1 (Abbreviated)	Analyze the following image-text pairs, understand the task, and use this to generate the response with a new image.
Middle (Abbreviated)	After reviewing the above image-text pairs, analyze the task and use this understanding to generate the response with a new image.
End (Abbreviated)	After reviewing the above image-text pairs and a query with a new image, analyze the task and use this understanding to generate the response.

Table 4: Formats of different instruction types and their corresponding details used in the prompt structure for all VL tasks. (Abbreviated) means that the instruction is a simplified version produced by GPT-4o.

Datasets	Training	Validation	Test	$\hat{D}$ Size
VQAv2	443,757	214,354	447,793	8000
VizWiz	20,523	4,319	8,000	2000
OK-VQA	9,055	5,000	/	800
Flickr30k	29,783	1,000	1,000	2500
MSCOCO	82,783	40,504	40,775	3000
HatefulMemes	8,500	500	2,000	800
Hybrid	30000	9000	/	3000
Fast	5,000	/	200	500
CLEVR	800	/	200	80

Table 5: Overview of the size distribution across the datasets used.

include special tags at the beginning and end of the prompt.

Each model, including OpenFlamingov2, Idefics2, InternVL2.5, and Qwen2.5VL, employs a structured approach to engage with image-text pairs. The two-phase task requires LVLMs to first absorb information from a series of prompts before utilizing that context to answer subsequent questions related to new images. This method allows for enhanced understanding and reasoning based on prior knowledge and context, which is essential for accurate predictions in VL tasks.

E Experiment

E.1 Datasets and Models

E.1.1 Dataset

In our study, we explore various VL tasks that use diverse datasets to evaluate model performance. As illustrated in Figure 9, we use VQA datasets such as VQAv2, VizWiz, and OK-VQA, which test the models' abilities in question-answer scenarios. Additionally, we incorporate image captioning datasets such as Flickr30k and MSCOCO to assess descriptive accuracy, along with the HatefulMemes dataset for classification tasks focused on hate speech detection. This comprehensive approach allows us to thoroughly evaluate the models across different tasks. The size distribution of the training, validation and test sets in these VL datasets is shown in Table 5.

For the Open-ended VQA task, we utilize the following datasets: VQAv2, which contains images from the MSCOCO dataset and focuses on traditional question-answering pairs, testing the model's ability to understand both the image and the question. VizWiz presents a more challenging setting with lower-quality images and questions, along with a lot of unanswerable questions, push-

ing models to handle uncertainty and ambiguity. OK-VQA is distinct in that it requires the model to leverage external knowledge beyond the image content itself to generate correct answers, making it a benchmark for evaluating models' capacity to integrate outside information.

For the Image Captioning task, we use the Flickr30k and MSCOCO datasets. The Flickr30k dataset consists of images depicting everyday activities, with accompanying captions that provide concise descriptions of these scenes. The MSCOCO dataset is a widely-used benchmark featuring a diverse range of images with detailed and richly descriptive captions, ideal for evaluating image captioning models.

For the Image Classification task, we use the HatefulMemes dataset, which is an innovative dataset designed to reflect real-world challenges found in internet memes. It combines both visual and textual elements, requiring the model to jointly interpret the image and the overlaid text to detect instances of hate speech.

VL-ICL Bench covers a number of tasks, which include diverse multimodal ICL capabilities spanning concept binding, reasoning or fine-grained perception. Few-shot ICL is performed by sampling the ICDs from the training split and the query examples from the test split. We choose two image-to-text generation tasks from it, which reflects different key points of ICL. Fast Open MiniImageNet task assigns novel synthetic names (e.g., dax or perpo) to object categories, and LVLMs must learn these associations to name test images based on a few examples instead of their parametric knowledge, emphasizing the importance of rapid learning from ICDs. CLEVR Count Induction asks LVLMs to solve tasks like *"How many red objects are there in the scene?"* from examples rather than explicit prompts. The ICDs' images are accompanied by obscure queries formed as attribute-value pairs that identify a specific object type based on four attributes: size, shape, color, or material. Models must perform challenging reasoning to discern the task pattern and generate the correct count of objects that match the query attribute.

The datasets in our experiments are evaluated using task-specific metrics, as summarized in Table 6. For the VQA tasks, **Hybrid** dataset and tasks in VL-ICL Bench, we use accuracy as the metric to assess the models' ability to provide correct answers.
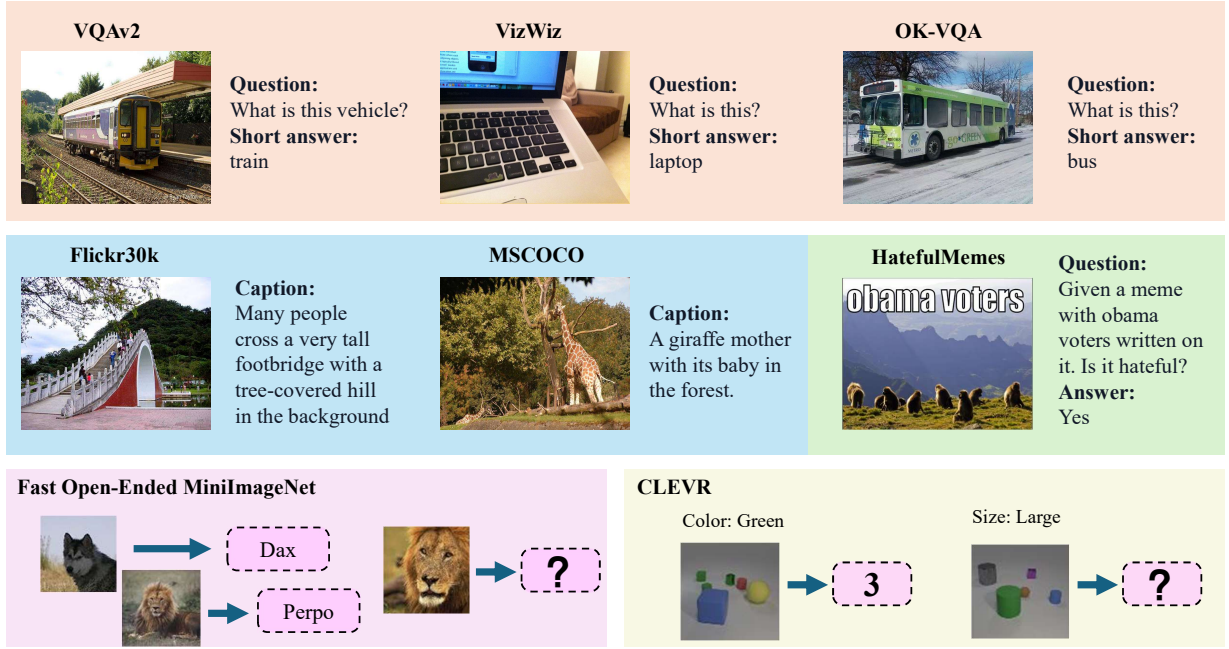
Figure 9: Illustrative examples from various vision-and-language datasets categorized by task type. Visual Question Answering (VQA) tasks are shown in red (VQAv2: train, VizWiz: laptop, OK-VQA: bus). Captioning tasks are represented in blue (Flickr30k: footbridge, MSCOCO: giraffes), while classification tasks are highlighted in green (HatefulMemes: meme identified as hateful). The bottom section demonstrates reasoning tasks with synthetic datasets: Fast Open-Ended MiniImageNet and CLEVR, focusing on conceptual understanding (e.g., assigning labels like "Dax" or identifying object properties like color and size).

Datasets	VQAv2	VizWiz	OK-VQA	Flickr30k	MSCOCO	HatefulMemes	Hybrid	Fast	CLEVR
metrics	Accuracy	Accuracy	Accuracy	CIDEr	CIDEr	ROC-AUC	Accuracy	Accuracy	Accuracy

Table 6: Evaluation metrics used for each benchmark. Accuracy is used for VQA datasets (VQAv2, VizWiz, OK-VQA), self-built **Hybrid** dataset, and two tasks in VL-ICL Bench. CIDEr (Vedantam et al., 2015) is used for image captioning datasets (Flickr30k, MSCOCO). ROC-AUC is used for the HatefulMemes classification task.

For the image captioning tasks, we use the CIDEr score, which measures the similarity between generated captions and human annotations. Finally, for the HatefulMemes classification task, we evaluate performance using the ROC-AUC metric, which reflects the model's ability to distinguish between hateful and non-hateful content.

E.1.2 LVLMs

In recent advances of LVLMs, efficient processing of multimodal inputs, especially images, has become a critical focus (Luo et al., 2024b; Li et al., 2024a, 2025b; Gu et al., 2025). Models like Open-Flamingov2, Idefics2, InternVL2.5, Qwen2.5VL, and GPT-4V implement unique strategies to manage and process visual data alongside textual input (Chen et al., 2025; Pan et al., 2025; Ni et al., 2023; An et al., 2024).

OpenFlamingov2 handles visual input by dividing images into patches and encoding them with a Vision Transformer. Each image patch generates a number of visual tokens, which are then processed alongside text inputs for multimodal tasks. To manage multi-image inputs, the model inserts special tokens <image> and <|endofchunk|> at the beginning and end of the visual token sequences. For example, an image divided into 4 patches produces 4 x 256 visual tokens, with the additional special tokens marking the boundaries before the tokens are processed by the large language model.

Idefics2 processes visual input by applying an adaptive patch division strategy adapted to image resolution and content complexity. Depending on these factors, each image is segmented into 1 to 6 patches, striking a balance between preserving spatial information and maintaining efficiency. These patches are encoded through a Vision Transformer, followed by a spatial attention mechanism and a

compact MLP, resulting in 128 visual tokens per patch. The positions of images in the input sequence are marked with <|image_pad|> for alignment, while <end_of_utterance> tokens separate query and answer components in in-context demonstrations. An image split into five patches yields 5 x 128 + 2 tokens before being integrated with the LLM.

InternVL2.5 dynamically divides each input image into tiles by selecting the closest aspect ratio i/j from a predefined set and resizing the image to S×i by S×j (with S = 448) before splitting it into i×j non-overlapping 448×448 px patches. Each patch is then fed through the InternViT encoder (InternViT-300M or InternViT-6B) to produce 1 024 patch embeddings, which are spatially downsampled via a pixel-unshuffle operation to yield exactly 256 visual tokens per patch. Special <img> and </img> tokens are inserted at the start and end of the full token sequence, so an image split into 3 patches produces 3 × 256 + 2 tokens before being passed to the LLM.

Qwen2.5VL reduces the number of visual tokens per image via an MLP-based merger that concatenates and compresses spatially adjacent patch features. A native-resolution ViT first splits an image (e.g. 224 × 224 with patch size 14) into a grid of patch embeddings. Rather than feeding all raw patches into the LLM, Qwen2.5VL groups each 2 × 2 block of adjacent patch features (four tokens), concatenates them, and projects the result through a two-layer MLP into a single fused token aligned with the LLM's embedding dimension. This achieves a 4× reduction in sequence length, dynamically compressing image feature sequences of varying lengths.

GPT-4V (Vision) extends GPT-4's capabilities to handle VL tasks by enabling the model to process and reason about visual input alongside text. The model can perform various tasks including image understanding, object recognition, text extraction, and visual question-answering through natural language interaction. In terms of its few-shot learning ability, GPT-4V demonstrates the capacity to adapt to new visual tasks given a small number of examples through natural language instructions, showing potential in areas such as image classification and visual reasoning, though performance may vary across different task domains and complexity levels.

E.2 Training Data Construction Details

We construct sequence data for model training using existing high-quality datasets, each corresponding to a VL task. The samples are uniformly formatted as (I, Q, R) triplets based on their respective task types. Each dataset generates a sequence set D_S for training, where each sequence consists of a query sample and N ICDs. The value of N is configurable, determining the number of shots during training. To ensure optimal training performance, we employ the same LVLM used in inference as a scorer to supervise the construction of D_S, making the method inherently model-specific. For each dataset, we construct D_S exclusively from its training set through the following three-step process: (1). We apply k-means clustering based on image features to partition the dataset into k clusters. From each cluster, we select the m samples closest to the centroid, yielding a total of $K = m \times k$ samples. These form the query sample set $\hat{D}$ after removing their ground-truth responses, which are stored separately in $D_{\hat{R}}$. The remaining dataset serves as the demonstration library DL. (2). For each query sample $\hat{x}_i \in \hat{D}$, we randomly sample a candidate set D_i of $64n$ demonstrations from DL. The objective is to retrieve N demonstrations from D_i that optimally configure the sequence for $\hat{x}_i = (\hat{I}_i, \hat{Q}_i)$ with its ground-truth response $\hat{R}_i = (\hat{R}_i^{(1)}, ..., \hat{R}i^{(t)})$. We use the log-likelihood score computed by the LVLM $\mathcal{M}$ as the selection criterion $\mathcal{CM}$, evaluating the model's predictive ability given a sequence with n ICDs:

$$\mathcal{C}_{\mathcal{M}}(S_i^n) = \sum_t log P_{\mathcal{M}}(\hat{R}_i^{(t)} \mid S_i^n, \hat{R}_i^{(1:t-1)}),$$
(28)

To determine the optimal n-th demonstration x_n for a sequence S_i^{n-1} with $n - 1$ ICDs, we select the candidate that maximizes the incremental gain in $\mathcal{C}_{\mathcal{M}}$:

$$x_n = \underset{x \in D_i}{argmax}[\mathcal{C}_{\mathcal{M}}(S_i^{n-1} + x) - \mathcal{C}_{\mathcal{M}}(S_i^{n-1})].$$
(29)

(3). We employ beam search with a beam size of $2N$, ensuring that for each $\hat{x}$, the top $2N$ optimal sequences are included in D_S. As a result, the final sequence set D_S consists of $2N \times k$ N-shot sequences, providing refined training data for the model.

E.3 Baselines

Various baseline methods are used to evaluate the model's performance, ranging from random sampling to different SOTA retrieval strategies. The following is a description of the baselines used in our experiments.

    1. **Random Sampling (RS)**: In this approach, a uniform distribution is followed to randomly sample n demonstrations from the library. These demonstrations are then directly inserted into the prompt to guide the model in answering the query.

    2. **Image2Image (I2I)**: During the retrieval process, only the image embeddings I_i from each demonstration $(I_i, Q_i, R_i$ are used. These embeddings are compared to the query image embedding $\hat{I}$ and the retrieval is based on the similarity between the images.

    3. **ImageQuery2ImageQuery (IQ2IQ)**: During the retrieval process, both the image embeddings I_i and the query embeddings Q_i of each demonstration $(I_i, Q_i, R_i$ are used. These embeddings are compared to the embedding of the concatenated query sample $(\hat{I}, \hat{Q})$, and the retrieval is based on the joint similarity between the images and the queries.

    4. **ImageQuery&Pseudo Response (IQPR)**: This baseline begins by using RS to generate a pseudo response $\hat{R}^P$ for the query sample. The pseudo response is concatenated with $\hat{I}$ and $\hat{Q}$ to create the query sample's complete embedding. We then retrieve $4n$ candidate samples from the dataset based on the similarity to this full embedding, and finally select the top n ICDs from these candidates using their Q–R similarity.

    5. **DEmO**: DEmO is a two-stage, data-free framework for configuring an optimal in-context sequence using influence, a concept that has become increasingly popular in ICD selection. In the first stage, it draws N random permutations $\{\pi_i\}_{i=1}^{N}$ of the candidate support set and measures each permutation's label-fairness by computing its content-free entropy:

$$E(\pi) = -\sum_l P(y = l \mid C_\pi) \log P(y = l \mid C_\pi),$$
(30)

where C_π is the prompt constructed by π with a content-free token in place of the query. The top-K permutations with the highest $E(\pi)$ are retained as the candidate set Π.

    In the second stage, for each candidate $\pi \in \Pi$ and test input x_t, DEmO computes the influence

Datasets	Training	Validation	Test	$\hat{D}$ Size	metrics
Rule Learning	1600	-	150		exact match scores
Fast Counting	800	-	40		Accuracy

Table 7: Overview of Rule Learning and Fast Counting tasks.

score

$$I(x_t; \pi) = P\big(y^* \mid x_t, C_\pi\big) - P\big(y^* \mid C_\pi\big),$$
$$y^* = \arg \max_y P\big(y \mid x_t, C_\pi\big),$$
(31)

which quantifies how much adding x_t shifts the model's confidence in its most likely label.

    Finally, permutation $\pi^* = \arg \max_{\pi \in \Pi} I(x_t; \pi)$ is chosen for the actual prediction. This targeted re-ranking ensures that each test sample uses the example order most "influential" to its correct classification, without relying on any additional labeled data.

    6. **Lever-LM**: Lever-LM is designed to capture statistical patterns between ICDs for an effective ICL sequence configuration. Observing that configuring an ICL sequence resembles composing a sentence, Lever-LM leverages a temporal learning approach to identify these patterns. A special dataset of effective ICL sequences is constructed to train Lever-LM. Once trained, its performance is validated by comparing it with similarity-based retrieval methods, demonstrating its ability to capture inter-ICD patterns and enhance ICL sequence configuration for LVLMs.

E.4 Results and Analysis

We can go deep into the per-model results in Table 8. The findings are as follows: (1) TACO exhibits the best performance in all but three tasks across nine datasets and five LVLMs, demonstrating its great efficiency and generalization. Upon examining the outputs, we observe that GPT-4V tends to deviate from the ICD format and produce redundant information more easily than open-source LVLMs, aligning with (Wu et al., 2023a). This results in the quality improvement of the ICL sequence not always translating into stable ICL performance gains for GPT-4V, which may explain why TACO did not achieve the best performance in two of its tasks. (2) For tasks like VizWiz and **Hybrid**, TACO consistently improves the quality of sequence generation in all LVLMs compared to similarity-based models, demonstrating the importance of increasing task semantics for complex task

Model	Method	VQA			Captioning		Classification	Hybrid	Fast	CLEVR
		VQAv2	VizWiz	OK-VQA	Flickr30K	MSCOCO	HatefulMemes			
OpenFlamingov2	RS	50.84	27.71	37.90	76.74	92.98	64.75	13.48	57.69	21.60
	I2I	49.52	26.82	37.79	79.84	94.31	69.53	12.79	59.07	19.39
	IQ2IQ	52.29	31.78	42.93	79.91	94.40	68.72	24.93	58.96	20.03
	IQPR	53.38	30.12	41.70	80.02	96.37	69.16	28.71	57.32	21.84
	DEmO	51.34	32.09	42.88	81.25	95.70	65.87	25.97	58.49	20.69
	Lever-LM	55.89	33.34	43.65	83.17	98.74	72.70	32.04	59.41	22.67
	Ours	**61.12**	**39.76**	**47.28**	**84.23**	**99.10**	**75.09**	**35.17**	**60.25**	**24.80**
Idefics2	RS	54.97	32.92	40.01	82.43	99.61	69.31	15.65	54.72	35.14
	I2I	53.77	31.67	41.37	85.76	101.34	69.64	10.49	55.20	32.37
	IQ2IQ	55.41	34.31	43.13	85.63	101.45	70.78	30.36	55.14	32.75
	IQPR	55.32	33.74	42.76	87.65	103.57	62.18	24.03	55.18	36.29
	DEmO	54.01	35.12	42.87	87.83	104.31	68.52	23.76	54.09	37.13
	Lever-LM	56.78	34.10	43.27	88.01	105.62	71.33	30.14	55.83	38.97
	Ours	**59.41**	**38.32**	**48.35**	**90.41**	**107.04**	**73.68**	**33.25**	**57.21**	**40.21**
InternVL2.5	RS	66.73	56.54	59.85	102.37	119.26	73.82	19.03	75.79	58.82
	I2I	64.71	56.03	59.51	105.31	121.10	77.05	16.03	77.03	57.79
	IQ2IQ	68.92	57.86	64.19	105.33	124.36	79.95	40.82	79.35	56.48
	IQPR	70.01	58.19	67.58	106.52	125.73	81.20	42.39	79.41	60.42
	DEmO	69.58	56.37	68.64	105.85	123.94	82.16	41.79	78.62	56.37
	Lever-LM	72.61	59.45	70.28	106.32	**127.51**	82.04	45.77	80.72	62.08
	Ours	**74.82**	**62.73**	**73.05**	**109.16**	127.43	**84.72**	**47.39**	**81.61**	**64.15**
Qwen2.5VL	RS	68.59	54.37	62.38	105.26	126.32	80.41	23.58	73.26	56.48
	I2I	66.98	53.81	62.75	105.78	126.43	78.62	15.79	79.84	54.83
	IQ2IQ	68.85	55.87	65.37	106.07	127.95	79.89	43.28	79.57	57.06
	IQPR	70.28	57.92	66.28	106.57	128.42	81.96	47.38	78.82	57.37
	DEmO	69.47	58.06	66.75	105.92	129.01	79.53	46.73	77.61	54.28
	Lever-LM	70.06	59.16	68.72	107.35	132.48	83.42	54.47	**80.53**	60.47
	Ours	**73.26**	**63.35**	**70.11**	107.02	**134.07**	**85.48**	**58.83**	80.39	**62.52**
GPT-4V	RS	60.49	45.38	59.13	101.56	115.87	82.40	16.98	58.72	45.08
	I2I	56.48	47.19	56.27	103.41	110.68	85.17	18.35	62.31	43.41
	IQ2IQ	-	-	-	-	-	-	-	-	-
	IQPR	-	-	-	-	-	-	-	-	-
	DEmO	-	-	-	-	-	-	-	-	-
	Lever-LM	**65.31**	54.62	65.73	106.34	126.98	**84.81**	45.62	60.31	48.34
	Ours	65.16	**56.17**	**68.89**	**107.29**	**129.71**	83.96	**51.48**	**64.17**	**50.59**

Table 8: Detailed results of different methods across all tasks for the five LVLMs used in the evaluation, with all generated sequences being 4-shot. The highest scores are highlighted in **bold**. Our model achieves the best performance in all but four tasks, demonstrating its generalization and effectiveness.

mappings. We find that the performance gains from TACO are not directly related to the model's intrinsic ability on these tasks. Unlike simpler tasks like classification, for tasks with complex mappings, task semantics still has a significant impact, even when LVLMs exhibit strong few-shot learning abilities. This shows that models with strong ICL capabilities on certain tasks retain, and even strengthen, their ability to leverage task semantics, underscoring the value of improving ICL sequence quality.

From all the above experiments, we can conclude that TACO effectively constructs prompts to maintain a coherent global task mapping. In this mapping, latent task signals from each demonstration are effectively integrated. Consequently, LVLMs can extract and synthesize fine-grained, task-specific information. This indicates that the key to superior performance lies in the prompt's ability to align with the underlying task intent, enabling deeper reasoning and more accurate outputs. The results in Table 2 and the corresponding analysis in Section 5 further explain TACO's good performance. The decline in performance observed when these components are removed indicates that maintaining a cohesive global mapping from individual demonstrations is fundamental to enabling the model to leverage task-relevant features during inference. Moreover, dynamic encoding during the ICL sequence configuration helps preserve and optimize the task mapping in the autoregressive process, thereby enhancing prompt quality.

Efficiency analyses. ICL is widely adopted for its efficiency (Kang et al., 2025); therefore, efficiency was a primary focus in the design of TACO. TACO exhibits high efficiency during both training and inference. Firstly, TACO is a lightweight language model composed solely of a fusion module and four transformer decoder blocks. With only 140M parameters, its training cost is extremely low compared to LLMs. Moreover, owing to its specialized training objectives, TACO prioritizes data quality over sheer quantity, enabling effective training with a relatively small amount of high-quality data. Simultaneously, our approach to constructing training data is highly efficient. By leveraging LVLM for self-assessment, we significantly reduce the overhead associated with incorporating external metrics. For instance, when using CIDEr to construct training data for the image captioning task, the costs are nearly 9 times higher than those of

our current method. To further validate TACO's training efficiency, we compare its training time with that of 4-layer Lever-LM on the same training sets. Table 9 demonstrates that TACO achieves superior performance compared to the baseline while maintaining comparable training costs, and even requires less training time than the baseline on several datasets. This evidence substantiates the high training efficiency of TACO. Moreover, to test inference efficiency, we compare different methods' retrieval time—that is, the time required to construct a 4-shot ICL sequence from instances in a specific dataset for a given query sample. Table 10 proves that TACO achieves notable performance improvements without compromising inference efficiency, with its runtime remaining comparable to that of RS.

E.5 More VL Tasks

To further demonstrate the broad applicability of our method to more tasks, especially challenging VL tasks, we evaluate on two additional benchmarks: GQA (Hudson and Manning, 2019) and A-OKVQA (Schwenk et al., 2022). Both require multi-hop reasoning, offering a more rigorous assessment of the model's performance under complex task mappings.

As shown in Table 11, TACO achieves the highest average results on both GQA and A-OKVQA. Consequently, our method attains optimal performance across all 11 datasets, which thoroughly demonstrates its robustness and effectiveness.

E.6 Discussions about Oracle

At inference time, ground truth responses are unavailable, so Oracle cannot be applied directly. Here, we adapt the pseudo-response generation idea from IQPR to Oracle. First, we generate a pseudo response using either RS or I2I; next, we treat this pseudo response as Oracle's ground truth for greedy retrieval. This process yields two additional baseline configuration methods. As demonstrated in Table 12, using pseudo results for Oracle can amplify the method's inherent drawbacks. On VQAv2, Oracle (I2I) performs 0.28% lower than I2I; on VizWiz, Oracle (RS) is 1.11% lower than RS, and Oracle (I2I) experiences an even more significant performance loss, being 1.24% lower than I2I. On other datasets, the performance of these two methods is also unstable. Therefore, using pseudo results to guide Oracle is a viable, yet not effective

Method	VQAv2	VizWiz	OK-VQA	Flickr30K	MSCOCO	HatefulMemes	**Hybrid**	**Fast**	**CLEVR**
Lever-LM	9.96	4.72	2.95	5.13	5.56	2.67	6.37	2.08	1.54
TACO	10.33	4.69	3.04	5.21	5.53	2.85	6.46	1.92	1.41

Table 9: GPU hours (h) consumed during training by two models.

Method	VQAv2	VizWiz	OK-VQA	Flickr30K	MSCOCO	HatefulMemes	**Hybrid**	**Fast**	**CLEVR**
RS	0.367	0.209	0.195	0.271	0.348	0.187	0.361	0.142	0.083
IQPR	0.639	0.301	0.287	0.574	0.725	0.352	0.701	0.291	0.200
Lever-LM	0.392	0.234	0.204	0.293	0.354	0.195	0.383	0.149	0.089
TACO	0.387	0.227	0.200	0.280	0.356	0.190	0.375	0.147	0.085

Table 10: Average retrieval time (s) (4-shot) of different methods across all LVLMs.

Method	GQA	A-OKVQA
RS	49.56	41.20
I2I	48.74	41.31
Lever-LM	56.28	45.93
TACO	**57.62**	**47.80**

Table 11: Average 4-shot results of different ICL sequence configuration methods on GQA and A-OKVQA benchmarks.

alternative. The limitations of the Oracle-based methods underline TACO's practical utility.

F Additional Ablation Study

F.1 Input Embeddings

To investigate the impact of input embedding construction on ICL sequence configuration, we vary both the training method of the CLIP encoders and the adoption of the fusion module to evaluate TACO's performance under different settings. For the CLIP encoders, we explore three alternative methods: one involves freezing its parameters and adding an MLP adapter to its output, which is then trained; another involves fully training the entire encoder; and the third involves training only the last two layers. For constructing the embeddings of multimodal ICD tokens, we first experiment with direct concatenation without fusion modules:

$$e_i = E_I(I_i) + E_T(Q)_i + E_T(R_i) + r_i, \quad (32)$$

where r_i is a randomly initialized learnable component introduced into the embedding. Besides binary fusion, we examine a finer-grained ternary fusion module that assigns separate weights to control the

contributions of all three components I, Q and R:

$$e_i = f_I \cdot E_I(I_i) + f_Q \cdot E_T(Q_i) + f_R \cdot E_T(R_i), \quad (33)$$

where f_I, f_Q and f_R denote the weights computed using a softmax function applied to the linear transformations, ensuring their sum equals 1. Additionally, we apply regularization to the weights: $f_I^2 + f_Q^2 + f_R^2 \leq \theta$ to prevent excessive reliance on specific components.

The training approach for CLIP affects the feature representation of embeddings, which in turn influences TACO's ability to capture cross-modal details during sequence configuration. From Table 13 we observe that for tasks with intrinsic features like VQA and **Hybrid**, leaving the CLIP unchanged or only adding an adapter leads to significant degradation in the quality of the ICL sequence generation. In fact, even methods that only train the last two layers show a more noticeable performance gap compared to the current approach. This highlights that the output pattern of the third-to-last layer of the encoder is crucial for capturing core task features in multimodal ICD. When we replaced our current training method with one that fully trains CLIP, we did not observe a significant performance drop. This suggests that TACO's treatment of ICDs as tokens does not cause feature loss. In contrast, through task-aware attention, it enhances feature representation, helping mitigate the limitations of the embedding itself. Considering the high cost of training the entire encoders, current method is optimal.

As we point out in §2, it is important for the model to focus on fine-grained features within the two modalities for multimodal ICL. However, Table 13 shows that the use of a ternary fusion mechanism to obtain more refined embeddings actually

Method	VQAv2	VizWiz	OK-VQA	Flickr30K	MSCOCO	HatefulMemes	Hybrid	Fast	CLEVR
Oracle(RS)	61.17	42.27	52.98	95.82	112.71	75.62	20.72	66.38	45.73
Oracle(I2I)	57.93	41.86	51.82	97.64	113.93	75.28	18.39	65.92	40.96

Table 12: Results of two Oracle-based configuration methods across 9 benchmarks.

	VQAv2	MSCOCO	HatefulMemes	**Hybrid**	**Fast**	**CLEVR**
(CLIP Encoder)						
N/A	45.38	111.57	70.21	37.67	60.84	43.52
Adapter only	47.29	112.42	73.26	40.15	63.59	45.71
Fully training	**49.63**	115.21	**78.49**	**43.84**	66.64	**47.80**
Last two	46.58	114.48	74.52	40.25	66.13	46.73
Last three	48.95	**115.36**	78.04	43.76	**66.72**	47.54
(Fusion Module)						
+ Ternary fusion	49.57	114.72	80.37	43.65	67.48	48.07
+ Binary fusion	**52.07**	**119.47**	**80.59**	**45.22**	**68.73**	**48.45**

Table 13: Results of TACO with different input embedding configurations. (CLIP Encoder) section shows the results without adding fusion modules under various training methods for CLIP encoders. N/A indicates no training or modification. (Fusion Module) section presents the results with two fusion modules added on top of the encoders trained with the method of training the last three layers.

Instruction	VizWiz	MSCOCO	HatefulMemes	**Hybrid**	**Fast**	**CLEVR**
$Beginning1$	**52.07**	119.47	80.59	**45.22**	**68.73**	**48.45**
$Inst' \rightarrow Inst$	46.21	114.36	78.23	39.64	63.52	43.70
$Beginning2$	49.61	118.78	79.13	44.15	67.42	46.38
$Beginning3$	49.25	118.23	78.49	43.71	67.29	45.62
$Middle$	51.85	119.62	**80.62**	45.18	68.53	48.27
End	51.73	**119.67**	80.37	44.96	68.59	47.89

Table 14: Results of TACO with diverse instruction types. The highest scores are highlighted in **bold**. $Inst' \rightarrow Inst$ means using $Inst$ during the initialization of TG.

$Inst'$	$Inst$	VQAv2	VizWiz	OK-VQA	**Hybrid**
	$Beginning2$	62.24	49.18	58.77	42.53
$Beginning1$	$Beginning3$	61.37	49.26	57.30	42.07
	End	63.58	50.69	59.62	42.26
$Beginning2$		65.28	51.05	60.81	44.61
$Beginning3$	$Beginning1$	64.62	51.28	59.14	44.28
End		65.40	49.08	60.47	43.72

Table 15: Results of TACO under various $Inst'$-$Inst$ combinations. $Inst'$ represents the style used for initializing TG, while $Inst$ refers to the style actually incorporated into the prompt.

results in worse performance compared to binary fusion, likely due to insufficient parameter capacity in TACO.

F.2 Instruction

Sections 2.2 and 5 highlight the importance of $Inst$ in improving multimodal ICL performance. However, as shown in Table 14, using the original embedding of $Inst$ to initialize TG degrades TACO performance due to semantic redundancy from long text embeddings, which can cause TG deviation and hinder convergence.

We further examine how the style and relative position of the $Inst$ affect performance. Its placement within the prompt is particularly critical: in standard and most effective ICL settings, all ICDs are positioned immediately before the query sample. Consequently, varying the relative position of $Inst$ serves as a direct probe of positional effects within the ICL sequence. Two new styles are developed and placed at the beginning of the prompt, while the CoT-style $Inst$ is also tested between the ICDs and query sample, as well as at the end. Diverse prompt samples are provided in Appendix D.2. Table 14 shows that, although the position of $Inst$ in the prompt has only a minor overall effect on performance, placing $Inst$ at the beginning yields the greatest relative gain. However, its style significantly affects performance, with the CoT-style being the most effective. Meanwhile, results

in Table 15 indicate that when the instruction used for TG initialization and the one included in the prompt have different styles, TACO demonstrates greater robustness. Changes in the style of $Inst'$ not only result in minimal performance degradation but also lead to significantly smaller performance variations. In contrast, for LVLMs, changes in $Inst$ style cause noticeable performance gaps and a clear preference for specific styles. This indicates that the performance fluctuations caused by $Inst$ are primarily attributable to LVLMs rather than TACO itself. Thus, $Inst$ can be viewed as a special ICD, contributing high-level local task mapping that integrates into the LVLM's global task mapping.

F.3 Generalization Test

To demonstrate the generalization of TACO beyond image-to-text tasks, we evaluate its performance on NLP and text-to-image tasks. We first use the latest LLM ICL benchmark, ICLEval's (Chen et al., 2024a) Rule Learning part to construct a mixed-task NLP dataset and test it on Qwen2-7B and LLaMA3-8B. For text-to-image tasks, we use the Fast Counting dataset from the VL-ICL Bench and

Methods	NLP		text-to-image
	Qwen2-7B	LLaMA3-8B	Emu2-Gen
RS	0.29	0.30	43.67
Q2Q	0.48	0.54	47.83
QPR	0.47	0.56	49.06
Lever-LM	0.50	0.60	-
Ours	**0.52**	**0.61**	**51.18**

Table 16: Results of different ICL sequence configuration methods in NLP and text-to-image tasks. Both training and generated shots are set to 4. The highest scores are highlighted in **bold**.
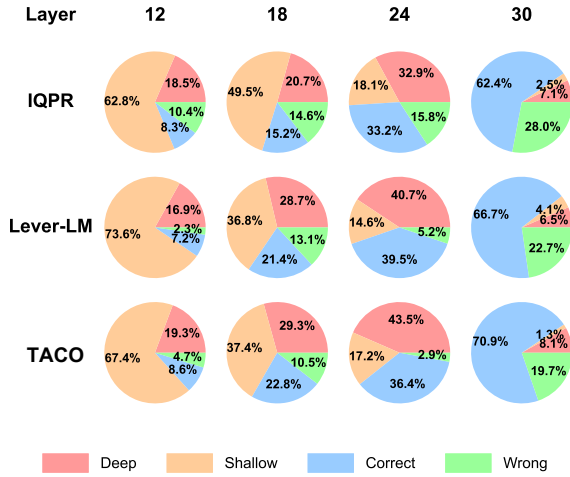

Figure 10: Visualization of the in-context lens for different methods under the Harder-Mapping setting.

test it on Emu2-Gen (Sun et al., 2024). The ICDs in both tasks can be represented as (Q, R). Results in Table 16 show that TACO consistently outperforms baselines across all tasks, highlighting its strong generalizability and wide application potential. For NLP evaluation, we utilize the Rule Learning part of the latest benchmark, ICLEval. ICLEval is designed to assess the ICL abilities of LLMs, focusing on two main sub-abilities: exact copying and rule learning. The Rule Learning part evaluates how well LLMs can derive and apply rules from examples in the context. This includes tasks such as format learning, where models must replicate and adapt formats from given examples, and order and statistics-based rule learning, where the model must discern and implement patterns such as item sequencing or handling duplications. These tasks challenge LLMs to go beyond language fluency, testing their ability to generalize from context in diverse scenarios. Examples of (Q, R) pairs can be found in Table 18. For all tasks, we use exact match scores to evaluate the predictions against the labels.

For text-to-image evaluation, we utilize the Fast Counting task in the VL-ICL Bench. In this task, artificial names are associated with the counts of objects in the image. The task is to generate an image that shows a given object in quantity associated with the keyword (e.g. perpo dogs where perpo means two). Thus, each Q is a two-word phrase such as "perpo dogs", and its corresponding R is an image of two dogs.

The ICDs in both tasks can be represented as (Q, R). In NLP, both Q and R are text; in text-to-image, Q is text while R is an image. We simply need to adjust the embedding encoder and fusion module accordingly. The baselines are RS, **Q2Q** (Query-to-query), **QPR** (Query&pseudo-response), and Lever-LM (not applicable to text-to-image).
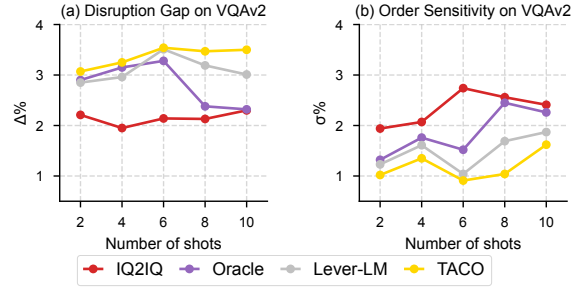

Figure 11: Analysis of task mapping cohesion in n-shot ICL sequences generated by different methods.

Method	HatefulMemes(Standard)	HatefulMemes(HM)
IQPR	73.62	68.31
Lever-LM	78.86	72.15
TACO	**80.59**	**74.87**

Table 17: Comparison of different methods under a Harder Mapping setting.

F.4 Revisiting Task Mapping Framework

In this section, we apply TACO's experimental results to the task mapping theoretical framework outlined in §2, thereby further validating its effectiveness and generality.

We first utilize the two metrics introduced in §2.3, Disruption Gap (Δ) and Order Sensitivity (σ), to evaluate task mapping cohesion in ICL sequences generated by TACO. Figure 11 shows that TACO achieves the highest Δ and lowest σ across all shots. This not only indicates that TACO-generated ICL sequences construct robust task mappings effectively utilized by LVLMs but also pro-

Task	Q	R
Format rules	|Index|name|age|city| |—|—|—|—| |1|Elijah Morgan|36|Pittsburgh|	\<person\> \<name\>Elijah Morgan\</name\> \<age\>36\</age\> \<city\>Pittsburgh\</city\> \</person\>
Statistics rules	588 and 823 are friends. 885 and 823 are friends. 795 and 588 are friends. 890 and 823 are friends. 885 and 588 are friends. 890 and 588 are friends. 795 and 823 are friends. Query: Who are the friends of 885?	823, 588
Order rules	Input: activity, brief, wonder, anger Output: anger, wonder, activity, brief Input: market, forever, will, curve Output: curve, will, market, forever Input: pain, leading, drag, shoot Output: shoot, drag, pain, leading Input: shopping, drama, care, start Output:	start, care, shopping, drama
List Mapping	Input: [1, 3, 6, 1, 83] Output: [3] Input: [5, 6, 35, 3, 67, 41, 27, 82] Output: [6, 35, 3, 67, 41] Input: [8, 45, 6, 18, 94, 0, 1, 2, 7, 34] Output: [45, 6, 18, 94, 0, 1, 2, 7] Input: [2, 7, 66, 6, 93, 4, 47] Output:	[7, 66]

Table 18: The examples of four Rule Learning tasks in ICLEval.

vides further evidence supporting the validity of our task mapping framework. Notably, from the results at shots 8 and 10, we observe that although TACO's training data is constructed by Oracle, it overcomes the cohesion weakening caused by bias accumulation through task mapping augmentation.

Next, we employ HatefulMemes's Harder-Mapping setting to evaluate TACO's performance on more challenging specific-mapping tasks. Results in Table 17 indicate that shifting task mapping from standard to harder reduces the performance of all three methods, but TACO still achieves the highest scores under both settings. Harder-Mapping setting increases the difficulty of understanding task mapping in ICDs, preventing the model from bypassing deep reasoning through parametric knowledge. In contrast, TACO guides LVLMs to gener-

ate ICL sequences with clearer, more identifiable global mappings, enabling them to overcome the comprehension barriers introduced by more challenging mappings.

Besides evaluating performance, we also employ in-context lens to examine the average evolution of the internal outputs in LVLM by these methods under HM settings. Figure 10 illustrates the evolution of the LVLM's internal reasoning as it learns from ICL sequences generated by different methods. The results show that sequences produced by TACO enable the model to most effectively infer task mappings and generate accurate responses.