How to Configure Good In-Context Sequence for Visual Question Answering

Supplementary Material

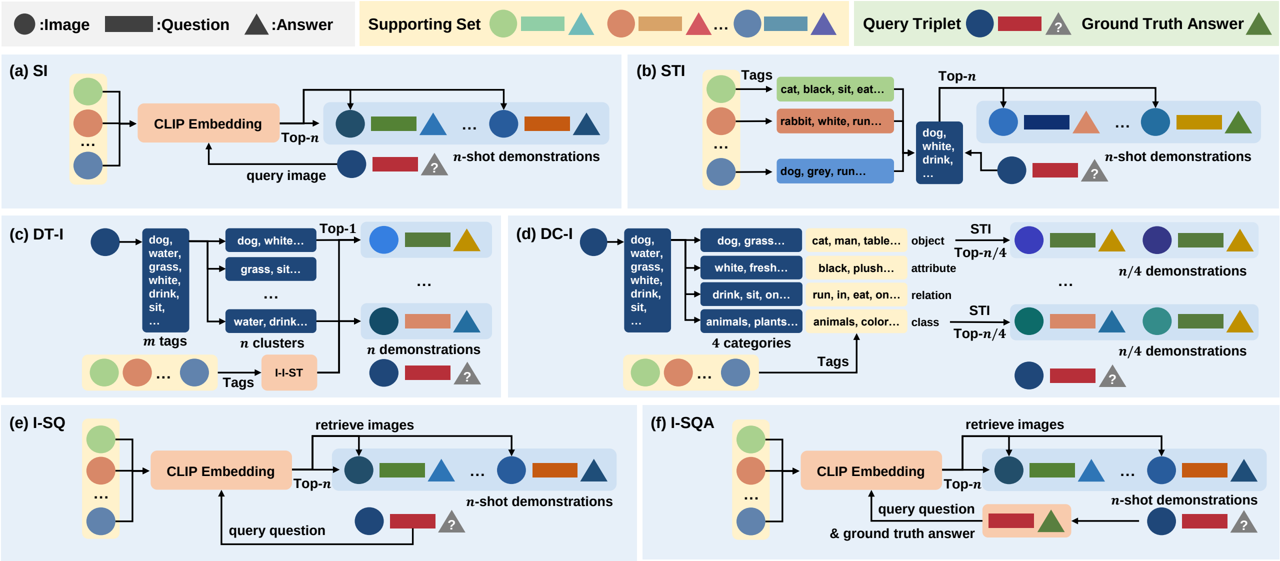


Figure 1. The schematic representation of more demonstrations retrieval strategies. (a) is Retrieving via Similar Image (SI) mentioned in Sec.3.2 in the main text of the paper, (b)-(f) is the image retrieval method mentioned in Sec. 1.1. Here we explore more specific retrieval methods, such as focusing on diversity in image retrieval ((c), (d)) and using another modality of information (text) for image retrieval ((e), (f)).

1. More Demonstrations Retrieval Methods

1.1. Retrieving Images

Here we introduce more methods centered around retrieving images from $\mathcal{D}$, subsequently using the corresponding triplet as the demonstrations.

(1) Retrieving via Similar Tags from Image (STI) (Fig. 1 (b)). We employ Vinvl [14] and IETrans [13] to extract three categories of tags (object, attribute, and relation) from the images in $\mathcal{D}$ and a given query image. Subsequently, we compute tag overlap between them, aiming to identify and return images from $\mathcal{D}$ exhibiting the highest similarity to the query. Consider a query image tagged with three tags:“dog”, “white” ,“drink”. Suppose image A is tagged with “cat”, “white”, “sit”, and image B with “dog”, “brown”, “drink”. STI would prioritize image B due to its higher tag overlap, having two matching tags with the query image. In order to efficiently calculate the overlap in tags, we list all discrete tags utilizing a one-hot manner and then apply an “AND” operation to assess the similarity at the tag level.

(2) Retrieving via Diverse Image (DI). Some works in NLP have discovered that diverse demonstrations containing more relevant information can significantly enhance performance. Drawing on this insight, we retrieve images from $\mathcal{D}$ based on diversity. We extract specific semantic labels from the images and apply two partitioning methods to divide the semantic labels into different clusters to meet the diversity of images for each cluster: **1) Retrieving via Diverse Tags from Image (DT-I)** (Fig. 1 (c)): we first extract three categories of tags (object, attribute, and relation) from $\hat{\boldsymbol{I}}$. Suppose there are total m kinds of tags, we divide them into n clusters and there are m/n kinds of tags in each cluster. Within each tag cluster, only the tags in that specific cluster are used to calculate the similarity score between two images, which is based on the number of tags both images share, using the SI method. **2) Retrieving via Diverse Categories from Image (DC-I)** (Fig. 1 (d)): we extract four categories of tags (object, attribute, relation, and class) from the images and categorize the tags into four respective clusters. For example, one cluster will solely contain object tags, while another may only encompass relation tags. This categorization process allows for a more holistic capture of all four content dimensions, retrieving the top-$\boldsymbol{n}/\mathbf{4}$ similar images within each clusters.

(3) Retrieving Image via Similar Text. In addition to using $\hat{\boldsymbol{I}}$ to retrieve images from $\mathcal{D}$, retrieving images based on the query text represents a method that can more effectively leverage the correlation between the visual modality and the linguistic modality. Considering that the image encoder and text encoder of CLIP can respectively map images and text into an embedding space, we employ the CLIP embedding of the query text to retrieve the corresponding embedding of images from $\mathcal{D}$. We propose two methods for text-based image retrieval: **1) Retrieving Image via Similar Question (I-SQ)** (Fig. 1 (e)): we evaluate the similarity between the CLIP embedding of the query question $\hat{\boldsymbol{Q}}$ and the CLIP embedding of each image $\boldsymbol{I}$ in $\mathcal{D}$. **2) Retrieving Image via Similar Question&Answer (I-SQA)** (Fig. 1 (f)): we use the CLIP embedding of the combination of $\hat{\boldsymbol{Q}}$ and the ground truth answer to retrieve images from $\mathcal{D}$ based on similarity metrics.

1.2. Retrieving Questions and Answers

(1) Retrieving via Similar Tags from Question (STQ). We extract different categories of tags from questions. Rather than leveraging the entire question sentence, we select pivotal tags for similarity retrieval. We adopt two types of settings: 1) Use the two most essential tags **(STQ-2)**: objects and relations. 2) Use four categories of tags **(STQ-4)**: objects, relations, attributes and interrogative words. Interrogative words are used to identify the type of questions, like "When".

(2) Retrieving via Diverse Question (DQ). Similar to DI, more diverse demonstrations enhance the comprehension and reasoning capabilities of the model. In addition to the vision level of diversity retrieval, we also perform diversity retrieval at the language level. We extract four categories of tags from questions: objects, relations, attributes and interrogative words. And we employ STQ to obtain the top-$n/4$ similar questions within each category.

(3) Retrieving Text via Similar Image. Similar to I-SQ and I-SQA, in addition to using text to retrieve questions from $\mathcal{D}$, we also use images to retrieve text, aiming to combine information from both visual and linguistic modalities. Specifically, we use $\hat{\boldsymbol{I}}$ to retrieve two types of text from $\mathcal{D}$: **1) Retrieving Question via Similar Image (Q-SI)**: we use the CLIP embedding of $\hat{\boldsymbol{I}}$ to retrieve the questions from $\mathcal{D}$. **2) Retrieving Question&Answer via Similar Image (QA-SI)**: we use the CLIP embedding of $\hat{\boldsymbol{I}}$ to retrieve the question-answer pairs $\{(\boldsymbol{Q}_1, \boldsymbol{A}_1); (\boldsymbol{Q}_2, \boldsymbol{A}_2); ...; (\boldsymbol{Q}_n, \boldsymbol{A}_n)\}$ from $\mathcal{D}$.

1.3. Manipulating Demonstrations

(1) Changing the Orders of Demonstrations. Some works in NLP have discovered the orders of the demonstrations can impact performance. Consequently, we also experiment with reversing the order of the demonstrations. We invert the original in-context sequence $\mathcal{S}$ to $\mathcal{S}' = \{(\boldsymbol{I}_n, \boldsymbol{Q}_n, \boldsymbol{A}_n); ...; (\boldsymbol{I}_2, \boldsymbol{Q}_2, \boldsymbol{A}_2)(\boldsymbol{I}_1, \boldsymbol{Q}_1, \boldsymbol{A}_1); (\hat{\boldsymbol{I}}, \hat{\boldsymbol{Q}})\}$.

(2) Changing Question-Answer Pairs into Declarative Sentences. Some works like [9] manipulate question-answer pairs into declarative sentences to better adapt to the pre-training language model. We follow [9] to manipulate the question-answer pairs, in which the corresponding short answer is replaced with a [MASK] token. For example, the question "How many animals are there?" can be changed into "There are [MASK] animals". Then we add the declarative sentences into the in-context sequence $\mathcal{S}' = \{(\boldsymbol{I}_1, \boldsymbol{D}_1, \boldsymbol{A}_1); ...; (\boldsymbol{I}_n, \boldsymbol{D}_n, \boldsymbol{A}_n); (\hat{\boldsymbol{I}}, \hat{\boldsymbol{D}})\}$, where $\boldsymbol{D}$ denotes the declarative sentence.

2. Experiments

2.1. More Details about Datasets and Evaluation Metric

VQAv2. VQAv2 [5] is a widely-used benchmark for visual question answering, consisting of 443,757 training and 214,354 validation questions. The dataset includes general images sourced from the MSCOCO dataset. Each image is associated with multiple questions and human-annotated answers, reflecting a combination of high-quality visual and commonly encountered questions.

VizWiz. The VizWiz dataset [4] is dedicated to answering visual questions from individuals who are blind. It consists of 20,523 training image/question pairs and 4,319 validation image/question pairs. Blind participants captured images and asked spoken questions, with 10 crowd-sourced answers per question. Due to this, the VizWiz dataset exhibits low-quality images and questions, also with a significant number of unanswerable questions.

OK-VQA. OK-VQA [10] aims to challenge models to leverage external knowledge for accurate answers. The dataset comprises 14,055 open-ended questions, each associated with five ground truth answers. All questions have been carefully filtered to ensure they necessitate external knowledge, such as information from Wikipedia.

VQA-CPv2. VQA-CPv2 [1] focuses on addressing the overfitting issue in Visual Question Answering (VQA) models through a change in the distribution of question-answer pairs between training and testing sets. Derived from the VQAv2 dataset, VQA-CPv2 is specifically designed to evaluate the robustness and generalization of VQA models by presenting them with a test set that significantly differs in the answer distribution for given question types compared to the training set. This ensures that models must rely on understanding the visual content and the question rather than exploiting dataset biases to perform accurately. To further investigate whether the same configuration in the experiment leads to the similar, we use this OOD(out-of-distribution) dataset to evaluate, the results and analysis is presented in Sec. 3.

Evaluation Metric. We follow [2] to use accuracy as the evaluation metric for VQA task. The detailed calculation

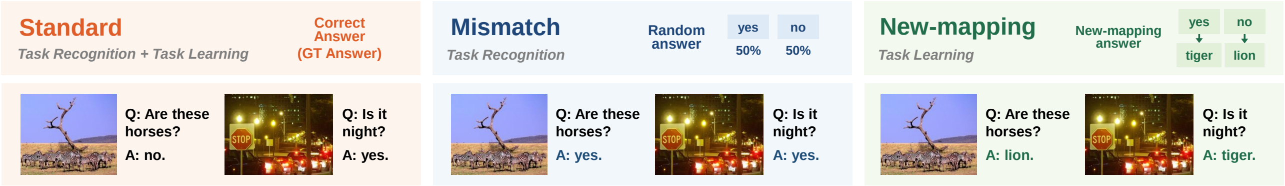


Figure 2. The experiments on disentangling TR and TL. Three settings are conducted on the data of VQAv2 whose answer types follow "yes/no" format: (1) "Standard" provides the correct demonstrations, preserving the TR and TL capabilities. (2) "Mismatch" randomly replaces answers to evaluate TR capability. (3) "New mapping" substitutes "yes/no" with novel answers (e.g., "tiger/lion") to test TL capability.

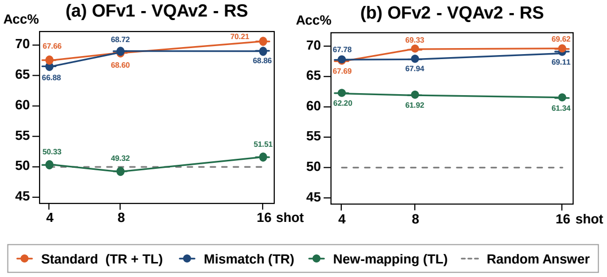


Figure 3. The results of the evaluation of the TR and TL abilities.

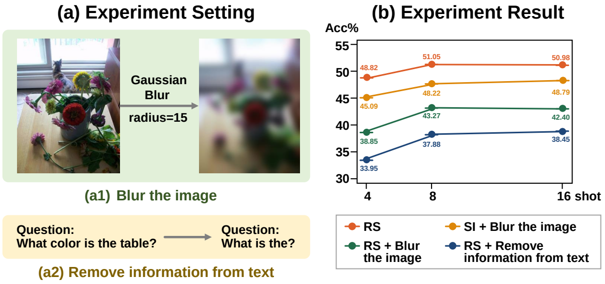


Figure 4. We add noise to the query image and question: (a1) using Gaussian Blur to blur the image, and (a2) removing information from the question. The experimental results are shown in (b).

formula is as follows:

$$Acc_{a_i} = min(1, \frac{3 \times \sum_{k\in[0,9]} match(a_i, g_k)}{10}), \qquad (1)$$

where a_i denotes the predicted answer of the LVLM, g_k denotes the k-th ground true answer, and the $match()$ function indicates whether two answers match, if they match, the result is 1, otherwise it is 0.

2.2. Implementation Details of Auxiliary Experiments

Disentangling TR and TL Following [12]. We conduct experiments on disentangling TR and TL following [12] as shown in Fig. 2 and Fig. 3. Specifically, we use different demonstration settings to reflect the TR and TL capabilities of models. We employ the standard in-context learning setting (**"standard"**) to represent the overall ICL (In-Context Learning) ability of the model, which involves the simultaneous application of TR and TL. Additionally, we use the Mismatching Answer (MA) method (**"mismatch"**), where the answers in the demonstrations are replaced with random answers that are 50% correct and 50% incorrect. This experiment aims to assess the TR ability of model, as format TR only requires the answer space to match the correct answer, while the mapping of incorrect answers significantly affects TL performance. Furthermore, we replace the answer in the demonstrations with an answer from a different answer space, forming a new mapping relationship (**"new-mapping"**). For example, replacing "yes" with "tiger" and "no" with "lion". This experiment aims to evaluate the TL ability of model, as such mapping relationships are rarely encountered during pre-training. To facilitate the replacement of the answer space, this experiment is conducted using only the "yes/no" type questions from the VQAv2 dataset.

Adding Noise to the Query Image and Question. To better compare the roles of visual information and textual information in TR, we conduct experiments adding noise to the query image and question, effectively diminishing the presence of pertinent information. Specifically, as shown in Fig. 5 and Fig. 4, we design two distinct experiments: **1) Adding Noise to the Image**: we apply Gaussian blur to the query image to blur the inherent visual information. As shown in Fig. 5 (b) and (c), the noise results in a loss of visual information, leading to errors in answers, while SI method can partially compensate for the loss of visual information. Although the answer is still incorrect in Figure 3 (c), the additional image information enables the model to acknowledge the potential presence of visual elements such as the "computer" in the original image, which can compensate for the visual TR ability to some extent. **2) Adding Noise to the Question**: we manipulate the query question by filtering out key information, such as nouns in the question which generally represent the object to be asked. Removing the key information may potentially hinder the model to understand what is being asked.

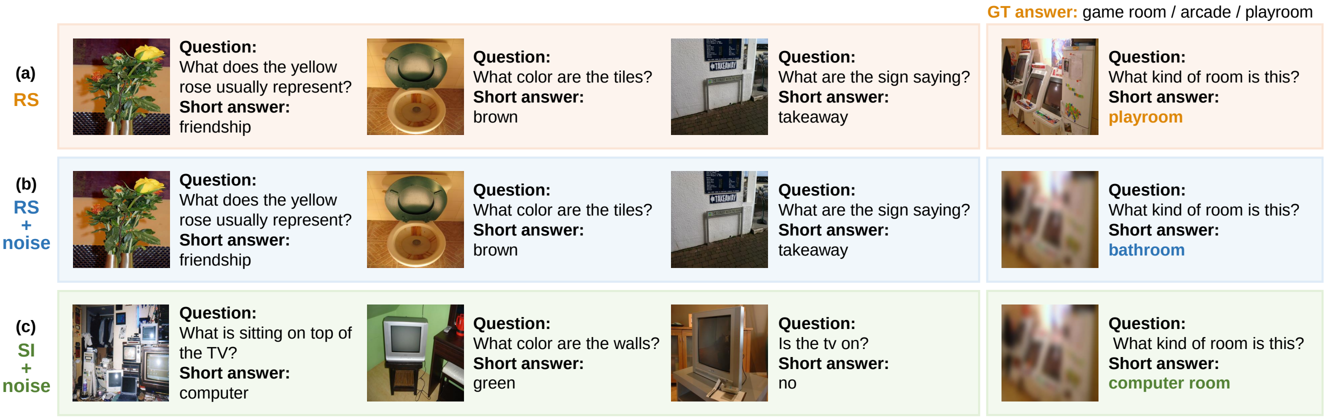

GT answer: game room / arcade / playroom
(a)
RS
Question: What does the yellow rose usually represent? Short answer: friendship
Question: What color are the tiles? Short answer: brown
Question: What are the sign saying? Short answer: takeaway
Question: What kind of room is this? Short answer: playroom
(b)
RS + noise
Question: What does the yellow rose usually represent? Short answer: friendship
Question: What color are the tiles? Short answer: brown
Question: What are the sign saying? Short answer: takeaway
Question: What kind of room is this? Short answer: bathroom
(c)
SI + noise
Question: What is sitting on top of the TV? Short answer: computer
Question: What color are the walls? Short answer: green
Question: Is the tv on? Short answer: no
Question: What kind of room is this? Short answer: computer room

Figure 5. The samples of the experiments on adding noise to the query image. (a) showcases a scenario where demonstrations are randomly sampled and no noise is added to the query image. The model correctly identifies the room as a "playroom" in response to the question. (b) depicts the use of identical demonstrations but with the query image blurred. The blurred image is difficult to provide effective visual information, resulting in incorrect answers. This error may be influenced by the second demonstration, which erroneously suggests a "bathroom" scene. (c) employs image similarity for demonstrations retrieval and blurs the query image. Although the answer is still wrong, the demonstrations in this instance provide relevant visual information, enabling the model to identify that this is in a "computer room".

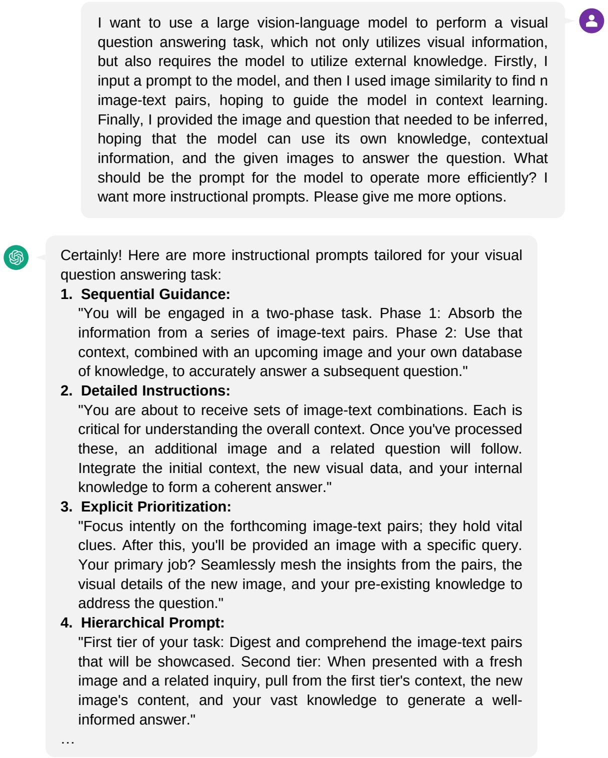
2.3. Implementation Details of Preparing Instructions

To better guide the generation of answers by LVLM, we attempt to add instructions to the input information. Besides using instructions written by humans, we also utilize instructions prompted by GPT-4 to further guide the LVLM. Considering that the VQA task on the OK-VQA dataset relies on external knowledge to answer questions, appropriate instructions might better stimulate the TL capability of the model. Therefore, our primary focus is to explore the impact of diverse instructions on performance for the OK-VQA dataset. Beyond manually written instructions, we have employed plowerful LLM, GPT4[11], to generate instructions. Specifically, we input our task description and instruction requirements to GPT4, asking it to output various instructions. We mainly require it to generate concise, straightforward instructions and detailed, hierarchical instructions. Fig. 6 displays the prompts we use and the responses from GPT-4.

3. More Experimental Results

Here, we present all the numerical results from the experiments, which are based on different demonstration retrieval and manipulation methods that are not mentioned in the main text. These methods are not discussed in the main text because they do not show significant improvement in the performance of OF or exhibit unstable results.

Tab. 1 showcases more similarity-based retrieval methods, including tag-based retrieval, which yields similar results to the embedding-based retrieval method mentioned in the main text. Both approaches exhibit stable improvements. However, using mixed modality for retrieval (I-SQ,



I want to use a large vision-language model to perform a visual question answering task, which not only utilizes visual information, but also requires the model to utilize external knowledge. Firstly, I input a prompt to the model, and then I used image similarity to find n image-text pairs, hoping to guide the model in context learning. Finally, I provided the image and question that needed to be inferred, hoping that the model can use its own knowledge, contextual information, and the given images to answer the question. What should be the prompt for the model to operate more efficiently? I want more instructional prompts. Please give me more options.

Certainly! Here are more instructional prompts tailored for your visual question answering task:

1. **Sequential Guidance:**
   "You will be engaged in a two-phase task. Phase 1: Absorb the information from a series of image-text pairs. Phase 2: Use that context, combined with an upcoming image and your own database of knowledge, to accurately answer a subsequent question."
2. **Detailed Instructions:**
   "You are about to receive sets of image-text combinations. Each is critical for understanding the overall context. Once you've processed these, an additional image and a related question will follow. Integrate the initial context, the new visual data, and your internal knowledge to form a coherent answer."
3. **Explicit Prioritization:**
   "Focus intently on the forthcoming image-text pairs; they hold vital clues. After this, you'll be provided an image with a specific query. Your primary job? Seamlessly mesh the insights from the pairs, the visual details of the new image, and your pre-existing knowledge to address the question."
4. **Hierarchical Prompt:**
   "First tier of your task: Digest and comprehend the image-text pairs that will be showcased. Second tier: When presented with a fresh image and a related inquiry, pull from the first tier's context, the new image's content, and your vast knowledge to generate a well-informed answer."

…

Figure 6. Scenario of using GPT4 for instructions generation . We first use concise language to describe our task, and then inform GPT4 of our requirements.

	OFv1			OFv2		
	4-shot	8-shot	16-shot	4-shot	8-shot	16-shot
RS	44.56	47.38	48.71	48.82	51.05	50.89
STI	44.61	47.89	49.91	50.02	51.83	51.89
I-SQ	44.22	47.18	49.29	47.14	50.47	50.96
I-SQA	43.84	46.58	48.31	47.20	48.43	49.29
STQ-2	48.58	50.01	51.47	47.53	49.28	49.42
STQ-4	49.74	51.16	52.58	48.69	50.18	48.83
Q-SI	43.07	45.56	47.33	46.54	49.12	48.56
QA-SI	45.24	47.83	49.14	47.52	49.65	50.28

Table 1. Experimental Results on the VQAv2 Dataset for more **Similarity-based Methods** in 4-shot, 8-shot, and 16-shot Learning Settings.

	OFv1			OFv2		
	4-shot	8-shot	16-shot	4-shot	8-shot	16-shot
RS	44.56	47.38	48.71	48.82	51.05	50.89
DT-I	46.44	48.18	49.86	49.27	51.75	51.09
DC-I	45.97	48.52	49.86	48.98	51.59	50.61
DQ	47.03	49.24	50.24	49.84	51.09	49.99

Table 2. Experimental Results on the VQAv2 Dataset for **Diversity-based Methods** in 4-shot, 8-shot, and 16-shot Learning Settings.

I-SQA, Q-SI, and QA-SI) does not provide significant assistance to the performance of model and may even lead to performance degradation. This could be attributed to the limited amount of image-related information present in the questions. Consequently, using images to retrieve questions or using questions to retrieve images is not a viable choice.

Tab. 2 displays the results of using diversity-based retrieval methods. It can be observed that enhancing the diversity of similar demonstrations can improve the results to some extent, but compared to similarity retrieval, it cannot yield significant improvements for OF.

Tab. 3 presents the impact of inputting similar demonstrations in reverse order to the model. Previous studies in the field of NLP [8] have found that placing more similar samples closer to the query can lead to greater performance improvements. However, based on the experimental results in this paper, such manipulation has little effect on OF.

Tab. 4 illustrates the effect of changing questions into declarative sentences. This transformation evidently leads to a marked reduction in performance of the model. One potential explanation for this phenomenon could be the difficulty for the model in recognizing the [mask] token. This challenge hinders the model to grasp the underlying reasoning requirements in the task. Consequently, this approach does not yield a significant improvement in OF either.

Tab. 5 presents the outcomes of offering different instructions on OK-VQA dataset. Both the concise instructions and the detailed, hierarchical instructions result in noticeable enhancements compared to the Random Sampling (RS) outcomes. Notably, concise and pertinent instructions appear to yield superior results. Building on these findings, we plan to delve deeper into the efficacy of different instructional methodologies.

Tab. 6 show the results on the out-of-distribution dataset VQA-CPv2. We find consistent conclusions, *e.g.*, mismatching image/answer (RS(MI)/RS(MA)) does not significantly hurt performance; using similar images and questions (SI-Q) can improve performance; and using instruction is effective.

4. Experiments on More Large Vision-Language Models

In our study, we primarily utilized Open-Flamingo, a Large Vision-Language Model (LVLM), as our experimental model. Despite sharing the same name, OFv1&v2 are acutally two different models since they use different LLMs (OFv1/OFv2 uses LLaMA/MPT) and they are trained on different data[3]. Similar findings across both models suggest a degree of generalizability. Additionally, we also explored other LVLMs that support in-context learning. For instance, we examine Otter[7], a LVLM based on Open-Flamingo, is fine-tuned on the MIMIC-IT multimodal dataset[6] for enhancing performance.

Furthermore, We compare the performance of these models trained on different large language models (LLMs). The primary distinction between Otter v1 and Otter v2 lies in their language models, similar to Open-Flamingo, Otter v1 utilizes LLAMA-7B, while Otter v2 employs MPT-7B. The outcomes of these comparative experiments are shown in Tab. 7.

	OFv1			OFv2		
	4-shot	8-shot	16-shot	4-shot	8-shot	16-shot
RS	44.56	47.38	48.71	48.82	51.05	50.89
SI	47.30	49.65	51.70	50.36	52.95	54.1
SI + Reverse	47.10	49.74	51.59	50.54	53.23	53.84
SQ	48.82	50.84	51.88	47.49	50.16	49.16
SQ + Reverse	47.75	49.79	50.80	47.38	48.77	47.74

Table 3. Experimental Results on the VQAv2 Dataset for **reversing the Order of Demonstrations** in 4-shot, 8-shot, and 16-shot Learning Settings.

	4-shot	8-shot	16-shot
RS	48.82	51.05	50.89
RS + Declarative Sentences	33.19	38.61	39.21
SI*	51.23	52.14	52.55
SI* + Declarative Sentences	41.89	44.04	44.11

Table 4. Experimental Results on the VQAv2 Dataset for **changing Question-Answer Pairs into Declarative Sentences** in 4-shot, 8-shot, and 16-shot Learning Settings. SI* denotes that we use SI to retrieve images and limit the retrieved images to be non repetitive.

References

[1] Aishwarya Agrawal, Dhruv Batra, Devi Parikh, and Aniruddha Kembhavi. Don't just assume; look and answer: Overcoming priors for visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4971–4980, 2018. 2

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022. 2

[3] Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023. 5

[4] Jeffrey P Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, et al. Vizwiz: nearly real-time answers to visual questions. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 333–342, 2010. 2

[5] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 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, 2017. 2

[6] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Fanyi Pu, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Mimic-it: Multi-modal in-context instruction tuning. *arXiv preprint arXiv:2306.05425*, 2023. 5

[7] Bo Li, Yuanhan Zhang, Liangyu Chen, Jinghao Wang, Jingkang Yang, and Ziwei Liu. Otter: A multi-modal model with in-context instruction tuning. *arXiv preprint arXiv:2305.03726*, 2023. 5

[8] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*, 2021. 5

[9] Yuhang Liu, Wei Wei, Daowan Peng, and Feida Zhu. Declaration-based prompt tuning for visual question answering. *arXiv preprint arXiv:2205.02456*, 2022. 2

[10] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 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, 2019. 2

[11] R OpenAI. Gpt-4 technical report. arxiv 2303.08774. *View in Article*, 2, 2023. 4

[12] Jane Pan. *What In-Context Learning "Learns" In-Context: Disentangling Task Recognition and Task Learning*. PhD thesis, Princeton University, 2023. 3

[13] Ao Zhang, Yuan Yao, Qianyu Chen, Wei Ji, Zhiyuan Liu, Maosong Sun, and Tat-Seng Chua. Fine-grained scene graph generation with data transfer. In *European conference on computer vision*, pages 409–424. Springer, 2022. 1

[14] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5579–5588, 2021. 1

Instruction	4 shot	8 shot	16 shot
RS	34.82	38.54	39.55
According to the previous question and answer pair, answer the final question.	35.72	39.38	40.46
Using external knowledge and image content to answer questions.	34.28	**40.47**	40.69
Integrate information from the question, image, and previous answers.	36.40	40.46	40.88
Consider the semantic relationship between the question and the image.	**36.45**	40.17	**41.11**
For the upcoming tasks, you'll be provided image-text pairs. Digest these pairs carefully. Later, an image along with a question will be presented. Combine your understanding from the pairs, the new image, and your own knowledge to answer.	35.13	40.30	40.61
You will be engaged in a two-phase task. Phase 1: Absorb the information from a series of image-text pairs. Phase 2: Use that context, combined with an upcoming image and your own database of knowledge, to accurately answer a subsequent question.	35.53	40.19	40.02

Table 5. Experimental Results of RS on the OK-VQA Dataset for **using different instructions** in 4-shot, 8-shot, and 16-shot Learning Settings.

	RS	RS(MI)	RS(MA)	RS(Instruction)	SI	SQ	SQA	SI-Q
(4&8-shot) Average	47.91	47.50	46.58	49.08	50.19	39.23	49.12	50.16

Table 6. Results of OFv2 on VQA-CPv2.

	LVLM	Language Model	4-shot	8-shot	16-shot	Average
RS	Open-Flamingo	LLAMA-7B	44.56	47.38	48.71	46.88
RS	Open-Flamingo	MPT-7B	48.82	51.05	50.89	50.25
RS	Otter	LLAMA-7B	39.14	41.28	42.43	41.13
RS	Otter	MPT-7B	24.96	27.60	29.92	27.49
SI	Open-Flamingo	LLAMA-7B	47.30	49.65	51.70	49.55
SI	Open-Flamingo	MPT-7B	50.36	52.95	54.10	52.47
SI	Otter	LLAMA-7B	37.61	39.72	41.42	39.58
SI	Otter	MPT-7B	23.08	24.38	23.95	23.80
SQ	Open-Flamingo	LLAMA-7B	48.82	50.84	51.88	50.82
SQ	Open-Flamingo	MPT-7B	47.49	50.16	49.16	48.94
SQ	Otter	LLAMA-7B	39.16	39.41	40.57	39.71
SQ	Otter	MPT-7B	23.29	23.56	22.74	23.20

Table 7. Experimental Results on the **different Large Vision-Language Models** in 4-shot, 8-shot, and 16-shot Learning Settings.

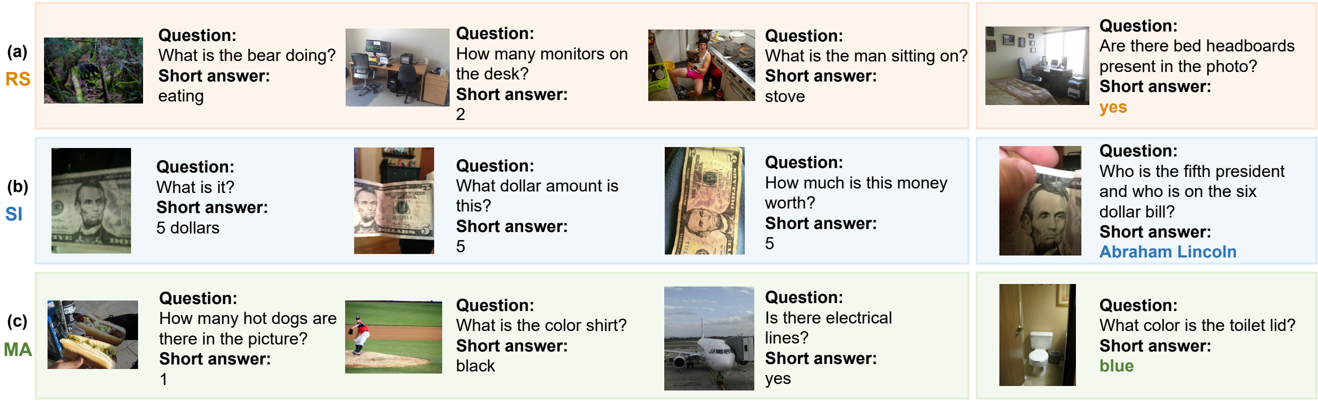

(a)
RS
Question: What is the bear doing? Short answer: eating
Question: How many monitors on the desk? Short answer: 2
Question: What is the man sitting on? Short answer: stove
Question: Are there bed headboards present in the photo? Short answer: yes
(b)
SI
Question: What is it? Short answer: 5 dollars
Question: What dollar amount is this? Short answer: 5
Question: How much is this money worth? Short answer: 5
Question: Who is the fifth president and who is on the six dollar bill? Short answer: Abraham Lincoln
(c)
MA
Question: How many hot dogs are there in the picture? Short answer: 1
Question: What is the color shirt? Short answer: black
Question: Is there electrical lines? Short answer: yes
Question: What color is the toilet lid? Short answer: blue

Figure 7. More input-output samples from our experiment of Open-Flamingo.