Do Vision-Language Models Learn In Context? Not So Fast

Anonymous ACL submission

Abstract

In-context learning enables Large Language Models (LLMs) to learn tasks from demonstration examples without parameter updates. While this ability has been extensively studied in LLMs, its effectiveness in Vision-Language Models (VLMs) remains underexplored. Existing research primarily focuses on a few models trained on interleaved image-text datasets and often overlooks image captioning in their analysis. In this work, we systematically analyze in-context learning in VLMs, evaluating six models across four architectures on three image captioning and four visual question answering benchmarks. We investigate the influence of prompt design, demonstration selection, model architecture, and training strategies. We also extend our analysis beyond models trained on interleaved datasets to include those trained on image-text pairs, often considered incapable of in-context learning. Our findings show that VLMs still struggle to leverage contextual information to adapt their outputs. However, detailed prompts specifying the task and structure of demonstrations improve performance more than simply concatenating examples. Additionally, while instruction-tuning enhances comprehension of detailed instructions, it reduces reliance on contextual examples and may hinder models' in-context learning capacity. Moreover, VLMs with advanced modality projectors can achieve competitive in-context learning performance even trained on image-text pairs.

1 Introduction

In recent years, Large Language Models (LLMs) have attracted significant attention for their notable performance across a wide range of Natural Language Processing tasks. As these models scale, in-context learning emerges as a new ability that allows LLMs to learn tasks given only a few examples through demonstrations (Brown et al., 2020; Wei et al., 2022). In this paradigm, before being asked to perform a task, the model is given a set of demonstrations, i.e., input-output examples, illustrating how to do it. Unlike supervised learning, in-context learning does not involve further parameter updates. Instead, the model should learn from analogy (Dong et al., 2024).

Despite the advancements, LLMs remain restricted to processing text-based data. They cannot handle other modalities such as image, audio, or video directly. However, the capacity to handle multimodal information contributes to knowledge acquisition and interaction with the real world. To bridge this gap, Vision-Language Models (VLMs) arise as a proposal to extend LLMs' capabilities to process visual information. Although in-context learning has been extensively studied in LLMs from various perspectives (Dong et al., 2024), relatively few works have explored this ability in VLMs (Baldassini et al., 2024; Qin et al., 2024; Yang et al., 2024). Moreover, they primarily evaluate a limited number of models trained on interleaved image-text datasets and focus predominantly on tasks such as Visual Question Answering (VQA) and image classification, often overlooking the task of image captioning.

In this paper, we systematically analyze in-context learning in VLMs, evaluating six models from four architectures across three image captioning and four VQA benchmarks. Specifically, we investigate how prompt construction, demonstration selection, and design decisions on model architecture and training impact in-context learning ability. Also, besides models trained on interleaved image-text datasets (OpenFlamingo (Awadalla et al., 2023) and Idefics2 (Laurençon et al., 2024)), we extend our analysis to include InstructBLIP (Dai et al., 2024) and LLaVA v1.5 (Liu et al., 2023), both originally designed to process a single image-text pair. To do so, we adapted their modality alignment method for multiple input images. We conduct all experiments in a controlled environment for fair comparisons, evaluating models under identical conditions.

Our main findings are as follows: (1) Overall, evaluated VLMs struggle to leverage the contextual information to adapt the output. However, using detailed prompts that explicitly define the task and the structure of demonstration examples proves more effective than simply concatenating examples. Additionally, increasing the number of demonstrations does not necessarily improve performance. (2) While instruction-tuning enhances the model's ability to comprehend detailed instructions, it may reduce its reliance on contextual examples. Conversely, training on interleaved image-text datasets improves the model's use of contextual information. (3) VLMs with advanced modality projectors achieve competitive in-context learning abilities even when trained on single image-text pairs, offering a cost-efficient alternative to models trained on large-scale interleaved datasets. In contrast, models with poor visual-text alignment – relying on long token sequences to represent images – show weaker in-context learning capabilities. These findings highlight crucial limitations in current VLMs that should be addressed to enhance their in-context learning ability.

2 Related Work

Vision-Language Models. VLMs excel in vision-language tasks due to pre-trained visual encoders and LLMs (Yin et al., 2024; Zhang et al., 2024). They comprise three key components: a visual encoder for image features, an LLM for text generation, and a modality projector to align visual and textual data, bridging the modality gap.

Various approaches have been explored for the modality projector, including linear layers and multi-layer perceptrons (MLPs) (Koh et al., 2023; Liu et al., 2023; Shukor et al., 2023; Su et al., 2023; Lin et al., 2024; Liu et al., 2024), which, despite the low training costs, can lead to long sequences of tokens thereby increasing the inference costs. Pooling strategies help mitigate this issue (Cha et al., 2024; Sun et al., 2024; Hu et al., 2024). Advanced methods like Q-Former (Li et al., 2023) improve alignment between frozen visual encoders and LLMs (Zhu et al., 2024a; Dai et al., 2024; Geigle et al., 2024). Another alternative is interleaved cross-attention layers (Alayrac et al., 2022; Laurençon et al., 2023; Xue et al., 2024), where the LLM directly attends to visual features but significantly increases the number of trainable parameters, as pointed out by Laurençon et al. (2024).

Training these models typically involves pre-training the modality projector on large-scale image-text datasets while keeping the visual encoder and LLM frozen for feature alignment. Subsequently, the LLM can be fine-tuned alongside the modality projector on instruction-following datasets to improve zero-shot generalization. Most works (Dai et al., 2024; Liu et al., 2024, 2023; Zhu et al., 2024a; Hu et al., 2024) train on a mixture of image captioning (Lin et al., 2014; Li et al., 2022; Sharma et al., 2018), VQA (Goyal et al., 2017; Schwenk et al., 2022; Marino et al., 2019), and instruction-following (Liu et al., 2024) datasets. Some models, such as Flamingo (Alayrac et al., 2022), Idefics (Laurençon et al., 2023; Laurençon et al., 2024; Laurençon et al., 2024), VILA (Lin et al., 2024), MMICL (Zhao et al., 2024), MM1 (McKinzie et al., 2025), and xGen-MM (BLIP-3) (Xue et al., 2024), are trained on interleaved image-text datasets (Laurençon et al., 2023; Zhu et al., 2024b) to further enhance multimodal reasoning capabilities.

In-Context Learning in VLMs. In-context learning has been extensively studied in LLMs, but this paradigm remains underexplored in VLMs. Recent studies investigate different factors that affect the in-context learning ability of VLMs, including modality importance, recency bias, demonstration retrieval, and ordering strategies. However, these studies primarily evaluate a limited number of models trained on interleaved image-text datasets, mainly in VQA and image classification tasks, often neglecting image captioning.

Yang et al. (2024) investigated in-context learning for image captioning, analyzing different demonstration retrieval and caption assignment methods. Their findings suggest that when demonstration images are similar to the query image, VLMs may leverage in-context captions as shortcuts to generate a new one rather than learning the captioning task.

Chen et al. (2024) and Baldassini et al. (2024) showed that textual information is more critical than visual information in the demonstrations for in-context learning in VLMs. Removing images causes a minor performance drop, while corrupting textual descriptions leads to a significant performance decline, indicating that VLMs heavily rely on textual cues even when processing multimodal demonstrations.

Beyond modality importance, Baldassini et al. (2024) explored recency bias in VLMs. They showed that models tend to replicate outputs from the most recent demonstrations, even when earlier demonstrations are more semantically relevant. Qin et al. (2024) further studied demonstration retrieval and ordering, revealing that multi-modal retrieval methods outperform single-modal approaches. They showed that the order of modalities within each demonstration can significantly influence model performance more than the arrangement of demonstrations themselves. Also, unlike traditional text-based in-context learning, where increasing the number of demonstrations improves performance, they found no significant performance gains when providing more demonstrations.

In contrast to previous studies, we systematically analyze the in-context learning ability of six models from four distinct architectures across three image captioning and four VQA benchmarks. We investigate the impact of prompt construction, demonstration selection, model architecture, and training choices. Additionally, previous works have explored models that support interleaved image-text inputs, in contrast, we modify InstructBLIP (Dai et al., 2024) and LLaVA v1.5 (Liu et al., 2023) to extend our analysis to models that originally designed for single image-text pairs.

3 Methodology

3.1 Experimental Setup

Models. We analyze four distinct families of VLMs: InstructBLIP (Dai et al., 2024), LLaVA v1.5 (Liu et al., 2023), OpenFlamingo (Awadalla et al., 2023), and Idefics2 (Laurençon et al., 2024). These families were selected to systematically explore how various design choices – such as bridging the modality gap and different training methods – affect the in-context learning capabilities of VLMs.

We use model checkpoints with parameter sizes ranging from 4B to 9B for a fair comparison across similar scenarios. Specifically, for InstructBLIP, we evaluate two checkpoints with different LLMs: InstructBLIP FlanT5-XL and InstructBLIP Vicuna 7B. For the other families, we assess LLaVA v1.5 7B, OpenFlamingo 9B, and two checkpoints of Idefics2 – before and after the instruction-tuning phase – namely, Idefics2 (Base) and Idefics2 (IT)[1].

Datasets & Metrics. We evaluate the models using different benchmarks proposed for image captioning and VQA. For image captioning, we use MS COCO (Lin et al., 2014), Flickr30K (Young et al., 2014) and NoCaps (Agrawal et al., 2019) datasets. We conduct our evaluation on the validation sets of each dataset. In image captioning experiments involving in-context learning, we utilize the MS COCO training set as the knowledge base from which we retrieve similar examples to construct the context. Each demonstration example comprises an image-text pair, where we randomly sample one of the human-annotated captions per image. We employ the CIDEr-D (Vedantam et al., 2015) and CIDEr-R (dos Santos et al., 2021), which are n-gram-based evaluation metrics, with CIDEr-R being less sensitive to variations in caption length.

For the VQA task, we utilize the VizWiz (Gurari et al., 2018), GQA (Hudson and Manning, 2019), TextVQA (Singh et al., 2019), and OKVQA (Marino et al., 2019) datasets, each designed to evaluate different model capabilities. VizWiz involves real-world images taken by visually impaired users with user-specific questions, while GQA assesses reasoning and compositional skills. TextVQA focuses on optical character recognition; thus, models should recognize text in images to answer the questions. OKVQA is designed to test models' ability to answer questions about images using external resources or commonsense knowledge. Unlike image captioning, we use each dataset's training set as the knowledge base. Performance is evaluated using the VQA accuracy metric (Antol et al., 2015), suitable for the open-ended nature of the questions.

3.2 Evaluation Protocol

Demonstrations Retrieval. Inspired by Yang et al. (2023), we retrieve demonstration examples employing a k-Nearest Neighbor approach based on the similarity distance in the visual feature space. We construct a knowledge base $\mathcal{D} = \{(i_1, t_1), \dots, (i_n, t_n)\}$, consisting of images i paired with their corresponding texts t different from those in the evaluation sets. Then, for each query image I, we extract its features $f(I)$ and we retrieve the top-k most similar image-text pairs based on the cosine similarity between visual features. Formally, the retrieved set

[1] Salesforce/instructblip-flan-t5-xl
Salesforce/instructblip-vicuna-7b
llava-hf/llava-1.5-7b-hf
openflamingo/OpenFlamingo-9B-vitl-mpt7b
HuggingFaceM4/idefics2-8b-base
HuggingFaceM4/idefics2-8b

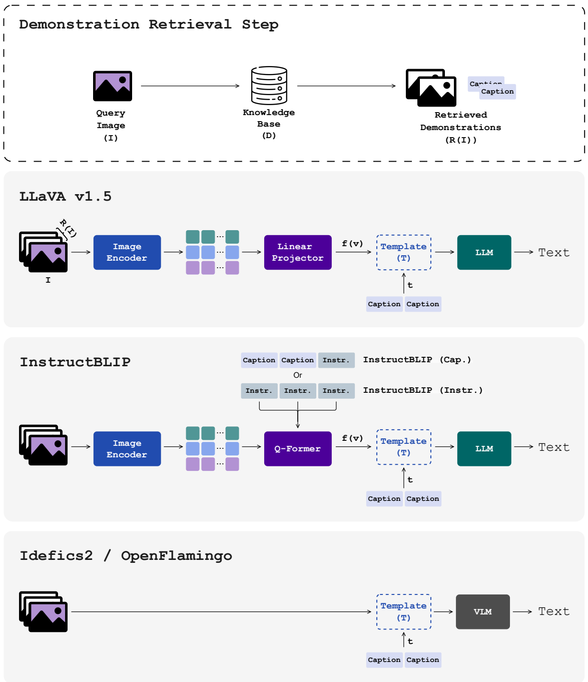

Figure 1: Evaluation pipeline for assessing the in-context learning capability of each analyzed model architecture. We illustrate the modifications made to the original LLaVA v1.5 and InstructBLIP pipelines to support interleaved image-text inputs.

Instruction-only Prompt:

<query image> A short image caption.

Straightforward Prompt:

<image$_1$> [Caption$_1$]
<image$_2$> [Caption$_2$]
<image$_3$> [Caption$_3$]
<query image> A short image caption.

Detailed Prompt:

I am an intelligent image captioning bot. Here are the features extracted by Q-Former for similar images along with their captions, following the format: [visual query tokens for example 1] [caption of example 1] ... [visual query tokens for example 3][caption of example 3] [visual query tokens for example 4]. Example 1: **<image$_1$>** [Caption$_1$], ..., Example 3: **<image$_3$>** [Caption$_3$].

<query image> A short caption I can generate to describe example 4 is:

Figure 2: Investigated prompt templates.

$\mathcal{R}(I)$ of image-text pairs is defined as $\mathcal{R}(I) = \{(i,t) \,|\, \text{top-}k_{(i,t)\in\mathcal{D}}\ sim(f_I, f_i)\}$[2], where $sim(\cdot)$ denotes the cosine similarity. We use a ViT (Dosovitskiy et al., 2021)[3] to encode the images. To investigate the impact of including multiple demonstration examples, we evaluate prompts containing 0, 1, 3, and 5 demonstrations.

In-Context Learning. We assess the in-context learning capabilities of the InstructBLIP, LLaVA, Idefics2, and OpenFlamingo architectures across various scenarios. Although in-context learning is straightforward for Idefics2 and OpenFlamingo, as they were trained with multiple interleaved image-text instances, implementing a similar pipeline for InstructBLIP and LLaVA poses some challenges. In Figure 1, we illustrate the pipeline adopted for each model architecture.

Since InstructBLIP and LLaVA were trained on image-text pairs, we adapted these models to handle multiple images per sample. Regarding LLaVA, for each sample, comprising multiple images interleaved with texts, we pass the images through the visual encoder and extract the visual features, which are then projected into the LLM token embedding space using an MLP block. Similarly, token embeddings are extracted for the texts. The projected visual features $f(v)$ and text embeddings t are concatenated into a single sequence and passed as input to the LLM.

In the case of InstructBLIP, we first extract the visual features for all images in the sample. However, unlike LLaVA, InstructBLIP employs an instruction-aware Q-Former to bridge modalities, which takes an image-text pair as input. This way, for the image captioning task, we explore two different approaches: (InstructBLIP Cap.) passing to the Q-Former the image-caption pairs for the demonstration examples, and the query image – for which we aim to generate the caption – alongside an instruction; and, (InstructBLIP Instr.) feeding Q-Former with image-instruction pairs for each image in the sample, including the query image. The output of the Q-Former is a set of query embeddings $f(v)$ that represent the visual information, with dimensions matching those of the LLM's input token embeddings. These query embeddings are, then, inserted into the template textual embeddings and fed into the LLM. For VQA, each demonstration example consists of an image and a corresponding question-answer pair, which are passed to the Q-Former. For the query image, we provide the image along with the question.

Prompt Construction. To evaluate the models' ability to adapt at inference time, we construct a

[2]For simplicity, we denote $f(i)$ as f_i and $f(I)$ as f_I.
[3]https://huggingface.co/google/vit-large-patch16-224-in21k

prompt by inserting the visual information $f(v)$[4] into a natural language template $\mathcal{T}$. We investigate scenarios using prompts with three different levels of detail, as illustrated in Figure 2. First, we use prompts containing only an instruction. Note that we do not necessarily use the same instructions as those reported in the original works. Instead, we choose to evaluate the different models under the same conditions. Next, we test straightforward prompts that include demonstration examples $\mathcal{R}(I)$ – image-caption pairs for image captioning and image-question-answer triplets for VQA – directly into the template $\mathcal{T}$. These examples are concatenated and followed by an instruction. Finally, building upon the Socratic Models (Zeng et al., 2023), we further explore detailed prompts based on Socratic templates (Zeng et al., 2023; Ramos et al., 2023) that specify the task and detail the format in which the demonstration examples are presented. In this case, the demonstrations are inserted at predefined positions within the template. We also experiment with minor variations of these templates to assess their impact. In all experiments involving demonstration examples, we follow the approach proposed by Baldassini et al. (2024), presenting examples in increasing similarity order relative to the query image as models tend to give more relevance to the last demonstrations. Specifically, we select the top-k examples, sorting them so that the most similar example is presented last.

4 Results and Discussions

Instruction-only Scenario. To establish a baseline and analyze the in-context learning capabilities of VLMs, we first conduct inference using instruction-only prompts without demonstration examples and investigate the VLMs' sensitivity to minor prompt variations. For this, we evaluate models on the image captioning task using four similar instructions, three sourced from (Dai et al., 2024): "`Write a short description for the image.`", "`A short image caption.`", and "`A short image description:`" along with a fourth variant, "`A short image description.`" where the colon is replaced with a period. Results in Table A1 show that InstructBLIP models (with Vicuna-7B and FlanT5-XL) exhibit consistent performance with minimal fluctuations, unlike other models. LLaVA demonstrates high sensitivity, with its performance on the MS COCO dataset declining significantly when the period in "`A short image description.`" is replaced with a colon, while remaining stable on other datasets. This suggests a potential memorization of MS COCO's content, as this dataset is used to generate instruction-following training data. In contrast, Idefics2 and OpenFlamingo perform best with "`A short image description:`" and show reduced performance when the colon is replaced with a period. Idefics2 (Base) exhibits greater variation before instruction-tuning, indicating that this phase enhances robustness to prompt variations.

Impact of Prompt on In-Context Learning. To investigate the influence of prompt structure on in-context learning, we evaluate models on the image captioning task using prompts designed with two levels of detail. The first prompt follows a straightforward template, where demonstration image-caption pairs are directly concatenated with an instruction. In contrast, the second prompt is more detailed, explicitly specifying the format in which the demonstration examples are presented and including the phrase "`I am an intelligent image captioning bot.`" (Section 3.2). For this experiment, we use the MS COCO training set as the knowledge base, and each sample includes three demonstration examples retrieved as context. The results of this evaluation, along with the best performance in the instruction-only scenario, are reported in Figure 3.

One can observe that all models, except instruction-tuned Idefics2, perform better in the instruction-only scenario than when provided with in-context demonstrations. These results indicate that these VLMs struggle to effectively utilize contextual information to adapt their outputs, thus exhibiting weak in-context learning abilities. Particularly, OpenFlamingo performs poorly with straightforward prompts, demonstrating a sharp decline in performance in this scenario. Furthermore, OpenFlamingo and instruction-tuned Idefics2, both of which are trained on interleaved image-text datasets, are the models least affected by the shift from instruction-only to in-context learning scenarios. It is worth noting that Idefics2 (Base) performs better with the straightforward prompt than with the detailed one. However, after instruction-tuning, its performance with the detailed prompt improves significantly, outperforming even the instruction-only

[4] The visual information $f(v)$ can consist solely of the query image, as in the instruction-only scenario, or also include the demonstrations $R(I)$, which is the case of the in-context learning.

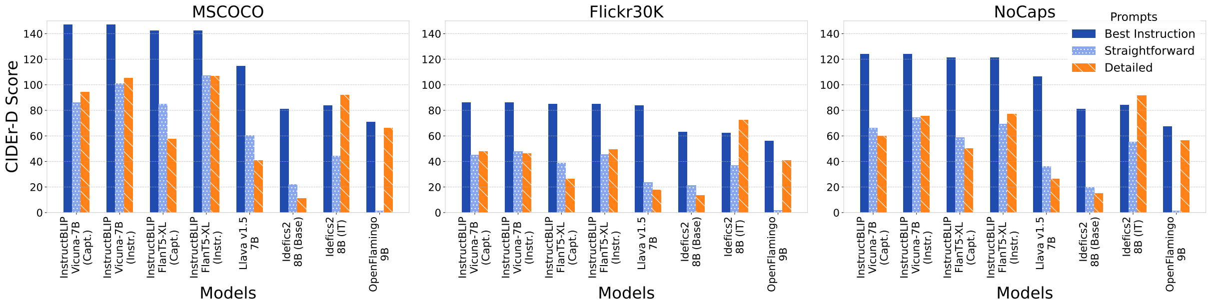

Figure 3: **Comparison of Instruction-only and In-Context Learning Scenarios**. Evaluation results for image captioning task under in-context learning using straightforward and detailed prompts. "Idefics2 8B (IT)" stands for the instruction-tuned checkpoint of the Idefics2 architecture.

setup, where its performance remains relatively stable. This result indicates that instruction-tuning enhances the model's ability to comprehend the detailed instruction, while training on interleaved image-text datasets helps the model better leverage contextual information.

Regarding InstructBLIP models, performance is further influenced by the type of input provided to the Q-Former. Specifically, using image-caption pairs from demonstration examples leads to lower performance than image-instruction pairs. Possibly, this is because InstructBLIP's Q-Former is primarily exposed to instructions rather than captions during instruction-tuning. Additionally, InstructBLIP FlanT5-XL performs better with straightforward prompts, whereas InstructBLIP Vicuna-7B achieves higher results with detailed prompts. This discrepancy is likely due to FlanT5-XL's fine-tuning on datasets containing few-shot exemplars, whose format is similar to the straightforward prompt.

Although there is a notable performance drop when shifting from instruction-only to in-context learning setup, InstructBLIP models remain competitive with Idefics2 and OpenFlamingo, despite not being trained on interleaved image-text datasets. In contrast, LLaVA struggles significantly in the in-context learning scenario. We hypothesize that Q-Former can compress the visual information into a small set of tokens, allowing InstructBLIP to better leverage the LLM's in-context learning ability. Conversely, LLaVA maps each visual patch into one input token embedding using a linear layer, requiring a long sequence of tokens to represent all input images (demonstrations and query), which may confound its LLM block. This hypothesis aligns with the findings of Laurençon et al. (2024), which suggest that reducing the number of visual tokens can improve performance on downstream tasks.

Overall, these results indicate that the evaluated VLMs struggle to leverage the contextual information and underscore the impact of prompt design on in-context learning performance. Detailed prompts that specify both the task and the structure of demonstration examples proved to be more effective than simply concatenating demonstrations. Also, our findings indicate that both instruction-tuning and training on interleaved image-text datasets enhance in-context learning ability. Notably, models with advanced modality projectors can achieve competitive performance even when trained on datasets containing only single image-text pairs per sample, offering a more cost-efficient alternative to training on interleaved datasets.

Influence of the Number of Demonstrations. In our previous experiments (Section 4), we fixed the number of demonstrations at three per sample. We observed that the detailed prompt generally improves performance. Building on this finding, we now investigate whether increasing the number of demonstrations (shots) in context further enhances model performance. To test this hypothesis, we evaluate the models on image captioning, using the previously defined detailed prompt, and on four VQA datasets. In this experiment, we vary the number of shots among 0, 1, 3, and 5. In the 0-shot setting, the prompt consists only of the template, without any demonstrations. We emphasize that this differs from the instruction-only scenario, as the 0-shot prompt signals a demonstration will be provided, but no actual demonstration is given. This setup allows us to evaluate the performance gains achieved by incorporating more demonstrations. The image captioning and VQA results are summarized in Figure 4 (the numeric results for image captioning and VQA can also be found in

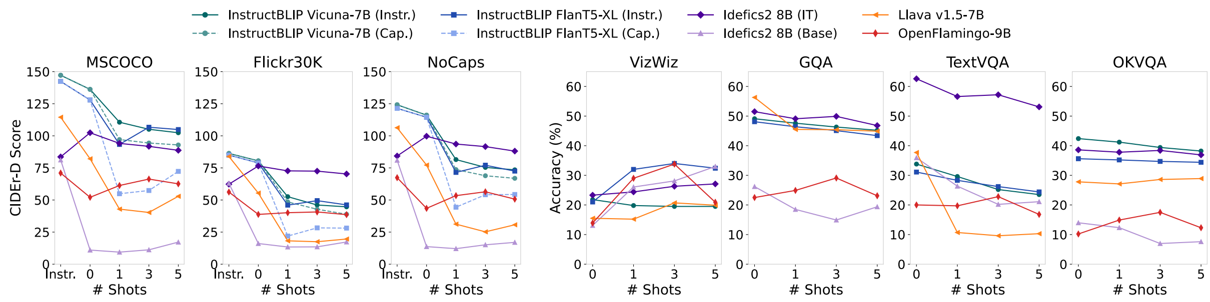

Figure 4: **Influence of the number of demonstration examples on performance**. We evaluate the impact of varying the number of demonstration examples (shots) in the context. For image captioning, we use a detailed prompt and employ the MS COCO training set as the knowledge base, plotting the CIDEr-D score. "Instr." in the x-axis of charts with image captioning results stands for the best results in the instruction-only scenario. For VQA, we utilize the corresponding training set for each dataset as the knowledge base and report the VQA accuracy.

Tables A3 and A4, respectively).

For image captioning, our results reveal that most models perform better in the instruction-only and 0-shot scenarios than when demonstrations are provided. Furthermore, we do not observe consistent improvements as demonstrations increase. In fact, incorporating more demonstrations often degrades performance relative to the 0-shot setup. However, consistent with prior observations, the Idefics2 and OpenFlamingo models appear to be the least affected by the demonstrations in the in-context learning setting. Specifically, Idefics2 (Base) and OpenFlamingo show slight performance gains as the number of shots increases, while the instruction-tuned Idefics2 model maintains a relatively stable performance. Note that InstructBLIP models achieve the highest performance on MS COCO but experience significant drops on Flickr30K and NoCaps, where the instruction-tuned Idefics2 model outperforms them. LLaVA is the most hampered by the demonstrations, it faces a notable decline on Flickr30K and NoCaps when demonstrations are included. This result corroborates our hypothesis that the long sequence of tokens required to represent the input images may confound the LLM.

Similar to image captioning, in VQA, we observe that models generally perform better across most datasets without in-context demonstrations. However, an opposite trend is observed for Vizwiz, where the inclusion of demonstrations appears beneficial. A detailed analysis (Section A.5.1) reveals that this effect is due to a strong dataset imbalance: the answer "unanswerable" appears more than a thousand times, while most other answers occur only once. Likewise, many of the provided demonstrations are also annotated as "unanswerable" leading models to favor this response. Additionally, in the TextVQA dataset, models' performance declines consistently as more demonstrations are introduced. This drop aligns with expectations, as answering questions in TextVQA requires recognizing text within images, and, in this case, similar examples in the context may confound the models.

Furthermore, for the GQA and OKVQA datasets, the performance of InstructBLIP models, LLaVA, and instruction-tuned Idefics2 remains relatively unchanged as the number of shots increases. This suggests that these models overlook in-context demonstrations for reasoning-based tasks. Nevertheless, it is interesting to note that they significantly outperform Idefics2 (Base) and OpenFlamingo on these datasets, underscoring the importance of instruction-tuning for VQA tasks requiring reasoning.

Our results suggest that increasing the number of demonstrations in the context does not necessarily enhance model performance. Instead, refining model architectures or training strategies may be necessary to leverage contextual information better. Particularly, instruction-tuned models achieve better results on reasoning-intensive VQA tasks, while models trained on interleaved image-text datasets exhibit better in-context learning ability. Due to computational constraints, our evaluation is limited to up to 5 demonstrations. However, our results show fluctuations in scores across 1, 3, and 5 shots. Therefore, further large-scale exploration is needed to fully understand the impact of number of demonstrations on performance.

Similar vs. Random Demonstrations. To investigate the impact of similar demonstrations on final results, we conduct a comparative analysis under two scenarios: one where demonstrations are similar to the query image and another with demonstrations from the same task but randomly chosen, either related or unrelated to the query image, referred to as random demonstrations. We hypothesize is that providing examples with content similar to the query image leads to better performance than using random demonstrations. To validate this, we fix the number of demonstrations at three and conduct experiments using both similar (as described in Section 3.2) and random demonstrations for image captioning and VQA tasks. For image captioning, we employ a detailed prompt to maintain consistency with previous experiments. Figure 5 illustrates the difference in scores between similar and random demonstrations across image captioning and VQA datasets.

Our experimental results highlight distinct behaviors across models in both image captioning and VQA tasks when exposed to similar and random demonstrations. In image captioning, InstructBLIP Vicuna-7B and LLaVA 7B demonstrate the most substantial performance gains with similar demonstrations, particularly on MS COCO and NoCaps datasets. In contrast, OpenFlamingo 9B exhibits a sharp performance drop, indicating that this model struggles to effectively leverage visual elements similar to the query image.

In VQA, most models benefit more from similar demonstrations than from random ones, with notable improvements on the OKVQA dataset. OKVQA consists of images and general questions that require commonsense knowledge. Then, similar demonstrations help models generate more accurate responses, whereas random demonstrations can confound them. In contrast, in TextVQA, models exhibit the greatest performance drop when using similar demonstrations. That is, models perform better with random demonstrations than with similar ones. We hypothesize that, as answering TextVQA questions requires recognizing text within images, showing random task-related examples might help models focus on the task itself. On the other hand, similar demonstrations could introduce visual distractions and lead to answer copying from provided examples.

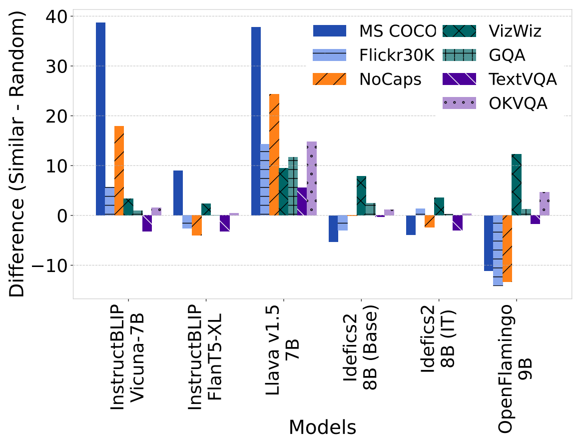

Figure 5: Difference in scores between in-context learning using similar demonstrations and random ones across image captioning and VQA datasets. For the image captioning datasets, we consider the detailed prompt. We plot the difference in CIDEr-D score for image captioning and VQA accuracy for VQA datasets.

5 Conclusion

In this paper, we systematically analyze in-context learning in VLMs, evaluating six models from four distinct architectures across multiple image captioning and VQA benchmarks. We investigate the impact of prompt construction, demonstration selection, and model design on in-context learning. Unlike previous work, we analyze models beyond those trained on interleaved image-text datasets. Our findings reveal that the evaluated models struggle to utilize contextual information to refine their outputs. However, detailed prompts, explicitly defining both the task and the structure of demonstration examples, significantly enhance this ability compared to simply concatenating examples. Increasing the number of demonstrations does not necessarily yield better results. While instruction-tuning helps models comprehend detailed instructions, it may reduce their in-context learning capacity. In contrast, training on interleaved image-text datasets enhances such ability. Additionally, we show that models with advanced modality projectors can achieve competitive in-context learning performance even when trained on single image-text pairs, offering a cost-efficient alternative.

This work sheds light on key limitations in current VLMs regarding their in-context learning ability. Future research should explore modality projectors to better integrate LLMs' in-context learning abilities into VLMs, as well as a combined approach using instruction-tuning and interleaved image-text training. Another promising direction is the inclusion of both positive and negative demonstrations, which could help models better distinguish between correct and incorrect responses.

Limitations

Although our analysis focuses on VLMs with up to 9B parameters and a maximum of 5 demonstrations per query due to computational constraints, studying larger models and increasing the number of shots would be important to determine whether our conclusions hold at a greater scale. Furthermore, to better understand the role of instruction-tuning and training of interleaved image-text datasets, it would be interesting to extend our analysis to a broader range of model architectures evaluating models before and after instruction-tuning. Finally, our analysis is limited to VLMs trained in English-language texts. However, evaluating the in-context learning capacity of multilingual models is essential. It would be necessary to study whether in-context learning can improve VLMs performance on low-resource languages.

Ethics Statement

This study systematically analyzes the in-context learning capabilities of publicly available VLMs. Our analysis is based solely on publicly available image captioning and VQA datasets, and we fully comply with the terms of use and licensing agreements associated with each model and dataset. We do not conduct any fine-tuning or modifications in the models that could introduce unintended risks. However, we recognize that our work reflects the existing limitations and potential risks of the evaluated models, including but not limited to gender, racial, and cultural biases, as well as the potential for generating misinformation or disinformation.

References

Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. 2019. NoCaps: Novel object captioning at scale. In *IEEE/CVF International Conference on Computer Vision*, pages 8948–8957.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikoł aj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. 2022. Flamingo: a visual language model for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 23716–23736. Curran Associates, Inc.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *IEEE International Conference on Computer Vision*, pages 2425–2433.

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

Folco Bertini Baldassini, Mustafa Shukor, Matthieu Cord, Laure Soulier, and Benjamin Piwowarski. 2024. What makes multimodal in-context learning work? In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1539–1550.

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

Junbum Cha, Wooyoung Kang, Jonghwan Mun, and Byungseok Roh. 2024. Honeybee: Locality-enhanced projector for multimodal llm. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13817–13827.

Shuo Chen, Zhen Han, Bailan He, Mark Buckley, Philip Torr, Volker Tresp, and Jindong Gu. 2024. Understanding and improving in-context learning on vision-language models. In *Workshop on Mathematical and Empirical Understanding of Foundation Models*.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2024. InstructBLIP: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36.

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

Gabriel Oliveira dos Santos, Esther Luna Colombini, and Sandra Avila. 2021. CIDEr-R: Robust Consensus-based Image Description Evaluation. In *Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, pages 351–360.

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

Gregor Geigle, Abhay Jain, Radu Timofte, and Goran Glavaš. 2024. mBLIP: Efficient Bootstrapping of Multilingual Vision-LLMs. In *3rd Workshop on Advances in Language and Vision Research (ALVR)*, pages 7–25, Bangkok, Thailand. Association for Computational Linguistics.

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 *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913.

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 *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3608–3617.

Anwen Hu, Haiyang Xu, Jiabo Ye, Ming Yan, Liang Zhang, Bo Zhang, Ji Zhang, Qin Jin, Fei Huang, and Jingren Zhou. 2024. mPLUG-DocOwl 1.5: Unified structure learning for OCR-free document understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 3096–3120, Miami, Florida, USA. Association for Computational Linguistics.

Drew A Hudson and Christopher D Manning. 2019. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6700–6709.

Jing Yu Koh, Ruslan Salakhutdinov, and Daniel Fried. 2023. Grounding language models to images for multimodal inputs and outputs. In *40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 17283–17300. PMLR.

Hugo Laurençon, Lucile Saulnier, Leo Tronchon, Stas Bekman, Amanpreet Singh, Anton Lozhkov, Thomas Wang, Siddharth Karamcheti, Alexander Rush, Douwe Kiela, Matthieu Cord, and Victor Sanh. 2023. OBELICS: An Open Web-Scale Filtered Dataset of Interleaved Image-Text Documents. In *Advances in Neural Information Processing Systems*, volume 36, pages 71683–71702. Curran Associates, Inc.

Hugo Laurençon, Leo Tronchon, Matthieu Cord, and Victor Sanh. 2024. What matters when building vision-language models? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Hugo Laurençon, Andrés Marafioti, Victor Sanh, and Léo Tronchon. 2024. Building and better understanding vision-language models: insights and future directions. *arXiv preprint arXiv:2408.12637*.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR.

Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. 2024. VILA: On Pre-training for Visual Language Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26689–26699.

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 *European Conference on Computer Vision*, pages 740–755.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2023. Improved baselines with visual instruction tuning. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2024. Visual instruction tuning. *Advances in Neural Information Processing Systems*, 36.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. OK-VQA: A visual question answering benchmark requiring external knowledge. In *IEEE/CVF Conference on Computer ision and Pattern Recognition*, pages 3195–3204.

Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Anton Belyi, Haotian Zhang, Karanjeet Singh, Doug Kang, Hongyu Hè, Max Schwarzer, Tom Gunter, Xiang Kong, Aonan Zhang, Jianyu Wang, Chong Wang, Nan Du, Tao Lei, Sam Wiseman, Mark Lee, Zirui Wang, Ruoming Pang, Peter Grasch, Alexander Toshev, and Yinfei Yang. 2025. MM1: Methods, Analysis and Insights from Multimodal LLM Pre-training. In *Computer Vision – ECCV 2024*, pages 304–323, Cham. Springer Nature Switzerland.

Libo Qin, Qiguang Chen, Hao Fei, Zhi Chen, Min Li, and Wanxiang Che. 2024. What factors affect multimodal in-context learning? an in-depth exploration. In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*.

Rita Ramos, Bruno Martins, and Desmond Elliott. 2023. Lmcap: Few-shot multilingual image captioning by retrieval augmented language model prompting. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1635–1651.

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.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia. Association for Computational Linguistics.

Mustafa Shukor, Corentin Dancette, and Matthieu Cord. 2023. eP-ALM: Efficient Perceptual Augmentation of Language Models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 22056–22069.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. 2019. Towards VQA models that can read. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8326.

Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. 2023. PandaGPT: One model to instruction-follow them all. In *1st Workshop on Taming Large Language Models: Controllability in the Era of Interactive Assistants!*, pages 11–23, Prague, Czech Republic. Association for Computational Linguistics.

Quan Sun, Yufeng Cui, Xiaosong Zhang, Fan Zhang, Qiying Yu, Yueze Wang, Yongming Rao, Jingjing Liu, Tiejun Huang, and Xinlong Wang. 2024. Generative multimodal models are in-context learners. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14398–14409.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. CIDEr: Consensus-based Image Description Evaluation. In *IEEE Conference on Computer Vision andPattern Recognition*, pages 4566–4575.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*.

Le Xue, Manli Shu, Anas Awadalla, Jun Wang, An Yan, Senthil Purushwalkam, Honglu Zhou, Viraj Prabhu, Yutong Dai, Michael S. Ryoo, Shrikant Kendre, Jieyu Zhang, Can Qin, Shu Zhang, Chia-Chih Chen, Ning Yu, Juntao Tan, Tulika Manoj Awalgaonkar, Shelby Heinecke, Huan Wang, Yejin Choi, Ludwig Schmidt, Zeyuan Chen, Silvio Savarese, Juan Carlos Niebles, Caiming Xiong, and Ran Xu. 2024. xGen-MM (BLIP-3): A Family of Open Large Multimodal Models. *CoRR*, abs/2408.08872.

Xu Yang, Yongliang Wu, Mingzhuo Yang, Haokun Chen, and Xin Geng. 2024. Exploring diverse in-context configurations for image captioning. *Advances in Neural Information Processing Systems*, 36.

Zhuolin Yang, Wei Ping, Zihan Liu, Vijay Korthikanti, Weili Nie, De-An Huang, Linxi Fan, Zhiding Yu, Shiyi Lan, Bo Li, et al. 2023. Re-ViLM: Retrieval-Augmented Visual Language Model for Zero and Few-Shot Image Captioning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11844–11857.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. 2024. A survey on multimodal large language models. *National Science Review*, pages 1–18.

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.

Andy Zeng, Maria Attarian, brian ichter, Krzysztof Marcin Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael S Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, and Pete Florence. 2023. Socratic models: Composing zero-shot multimodal reasoning with language. In *The Eleventh International Conference on Learning Representations*.

Duzhen Zhang, Yahan Yu, Jiahua Dong, Chenxing Li, Dan Su, Chenhui Chu, and Dong Yu. 2024. MM-LLMs: Recent advances in MultiModal large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12401–12430, Bangkok, Thailand. Association for Computational Linguistics.

Haozhe Zhao, Zefan Cai, Shuzheng Si, Xiaojian Ma, Kaikai An, Liang Chen, Zixuan Liu, Sheng Wang, Wenjuan Han, and Baobao Chang. 2024. MMICL: Empowering Vision-language Model with Multi-Modal In-Context Learning. In *The Twelfth International Conference on Learning Representations*.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2024a. MiniGPT-4: Enhancing Vision-Language Understanding with Advanced Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Wanrong Zhu, Jack Hessel, Anas Awadalla, Samir Yitzhak Gadre, Jesse Dodge, Alex Fang, Youngjae Yu, Ludwig Schmidt, William Yang Wang, and Yejin Choi. 2024b. Multimodal c4: An open, billion-scale corpus of images interleaved with text. *Advances in Neural Information Processing Systems*, 36.

A Appendix

A.1 Results on Instruction-only Scenario

As detailed in Section 4, we first conduct inference using instruction-only prompts, i.e., without including any demonstration examples, to establish a baseline for our in-context learning experiments. To do so, we test four similar instructions, three of which are selected from (Dai et al., 2024): (I1) "`Write a short description for the image.`", (I2) "`A short image caption.`", and (I3) "`A short image description:`". We also create a fourth instruction, (I4) "`A short image description.`", by replacing the colon in the latter instruction with a period. Table A1 summarizes the results of these experiments.

InstructBLIP models (with Vicuna-7B and FlanT5-XL) exhibit consistent performance, with only minor fluctuations across the different instructions. Interestingly, this consistency does not extend to the other models. LLaVA shows one of the greatest sensitivity to instruction variations, performing best with the instruction "`A short image caption.`" and worst with "`Write a short description for the image.`". Notably, its performance on the MS COCO dataset declines significantly when the period in "`A short image description.`" is replaced with a colon, while remaining stable on the other datasets. This drop in results on MS COCO suggests that LLaVA may be memorizing the content of MS COCO, as this dataset is used to generate instruction-following training data. In contrast, Idefics2 models and OpenFlamingo perform best with the instruction "`A short image description:`" and show reduced performance when the colon is replaced with a period. Also, the difference between the highest and lowest scores is more pronounced in Idefics2 before the instruction-tuning phase (Idefics2 (Base)), possibly because this phase enhances the model's robustness to minor prompt variations. A similar trend is observed in OpenFlamingo, which also does not undergo an instruction-tuning phase during training.

A.2 Experimental Results in Numbers

We provide the numerical results of the experiments regarding the impact of prompt in the in-context learning ability (Section 4) and the influence of the number of demonstrations in the context on the performance (Section 4). The results are divided into three tables. Table A2 presents the results for the image captioning task under instruction-only and in-context learning scenarios; it shows the best performance in the instruction-only scenario alongside the results of in-context learning with straightforward and detailed prompts. Tables A3 and A4 show the results for image captioning and VQA, respectively, varying the number of demonstration examples in the context.

A.3 Ablation on Detailed Prompt

Building on the findings from the instruction-only scenario (Section 4), we investigate the impact of small modifications to the detailed prompt used to evaluate the in-context learning capabilities of models in the image captioning task. We use *(1) Base detailed prompt* for our experiments and test various small changes to this template (Figure A1).

The results, summarized in Table A5, reveal interesting insights. First, removing the initial phrase (prompt 2) significantly hampers the performance of most models. Second, models generally perform better when the word "creative" is removed from the prompt (prompt 3). However, removing both the initial phrase and the word "creative" (prompt 4) produces intermediate results, suggesting that the effects of these changes are combined. The best prompt in most cases is to keep the initial phrase while removing the word "creative" (prompt 3), which leads to the highest performance.

These changes in the prompt can result in substantial performance differences, with variations of up to 20 points in CIDEr scores. Among the experimented models, InstructBLIP FlanT5-XL demonstrates major sensitivity to prompt modifications. Notably, it fails to generate captions when the word "creative" is included in the prompt, underscoring its dependence on precise prompt phrasing. Finally, as expected, altering the name of the modality projector (prompts 5, 6, and 7, Figure A1) has no impact on model performance, indicating that the models simply ignore this detail.

A.4 Impact of Training Size on In-Context Learning

To further explore the impacts of design decisions on in-context learning, we investigate the impact of the training set size on the model performance in both instruction-only and in-context learning scenarios. Figure A2 illustrates model performance across image captioning datasets as a function of training set size.

Table A1: **Instruction-only scenario**. We evaluate the VLMs on image captioning datasets with different instructions and report the CIDEr-D (↑) and CIDEr-R (↑) scores. The numbers in bold are at least 1 point better than the others. The evaluated instructions are: (I1) "`Write a short description for the image.`", (I2) "`A short image description.`", (I3) "`A short image description:`" and (I4) "`A short image caption.`".

Model	Instruction	MS COCO		Flickr30K		NoCaps	
		CIDEr-D	CIDEr-R	CIDEr-D	CIDEr-R	CIDEr-D	CIDEr-R
InstructBLIP Vicuna-7B	I1	147.4	149.5	85.1	97.0	123.7	130.2
	I2	146.7	148.4	85.9	97.8	124.0	130.0
	I3	146.8	149.0	86.3	98.4	123.7	130.5
	I4	147.2	149.0	86.3	98.2	124.2	130.5
InstructBLIP FlanT5-XL	I1	142.5	144.5	85.1	96.9	121.5	128.2
	I2	142.4	144.3	85.4	97.2	121.6	128.2
	I3	142.4	144.4	85.1	96.8	121.4	128.1
	I4	142.4	144.4	85.0	97.0	121.4	127.9
LLaVA v1.5-7B	I1	64.9	88.9	47.3	71.8	72.2	93.0
	I2	101.3	113.9	69.6	88.4	99.0	113.1
	I3	78.3	90.6	69.7	87.4	96.5	111.7
	I4	**114.5**	**122.7**	**83.9**	**99.2**	**106.3**	**117.5**
Idefics2-8B (Base)	I1	0.1	3.1	0.0	1.0	0.3	4.6
	I2	9.9	64.6	9.7	61.3	19.0	72.0
	I3	**81.2**	**94.6**	**63.0**	**79.7**	**81.0**	**95.3**
	I4	0.7	1.5	0.7	2.0	0.4	0.9
Idefics2-8B (Instruction-Tuned)	I1	57.5	70.1	51.8	66.5	69.1	80.9
	I2	49.1	59.6	47.5	61.8	67.7	79.5
	I3	**83.6**	**90.1**	**62.3**	**74.7**	**84.3**	**93.0**
	I4	53.5	65.3	41.9	55.9	63.2	75.8
OpenFlamingo-9B	I1	36.1	50.9	31.4	43.4	29.8	49.8
	I2	60.9	72.8	49.4	62.8	63.1	75.3
	I3	**71.0**	**82.0**	**56.2**	**69.8**	**67.4**	**81.5**
	I4	58.7	66.7	47.2	58.8	53.0	63.7

Table A2: **Comparison between instruction-only and in-context learning scenarios**. Evaluation results for image captioning task under in-context learning using straightforward and detailed prompts. "Instruction" refers to the best performance in the instruction-only scenario. Bold numbers highlight the best performance for each model.

Model	Prompt	MS COCO		Flickr30K		NoCaps	
		CIDEr-D (↑)	CIDEr-R (↑)	CIDEr-D (↑)	CIDEr-R (↑)	CIDEr-D (↑)	CIDEr-R (↑)
InstructBLIP Vicuna-7B (Q-Former fed with Caption)	Instruction	**147.2**	**149.0**	**86.3**	**98.2**	**124.2**	**130.5**
	Straightforward	86.2	96.0	45.0	52.1	66.1	73.3
	Detailed	94.4	97.9	47.8	55.8	60.0	71.3
InstructBLIP Vicuna-7B (Q-Former fed with Instruction)	Instruction	**147.2**	**149.0**	**86.3**	**98.2**	**124.2**	**130.5**
	Straightforward	100.9	106.3	47.7	53.6	74.2	78.5
	Detailed	105.1	107.5	46.1	52.3	75.4	79.0
InstructBLIP FlanT5-XL (Q-Former fed with Caption)	Instruction	**142.4**	**144.4**	**85.0**	**97.0**	**121.4**	**127.9**
	Straightforward	84.9	87.1	39.0	44.1	58.6	61.6
	Detailed	57.4	59.1	26.5	30.5	50.0	52.9
InstructBLIP FlanT5-XL (Q-Former fed with Instruction)	Instruction	**142.4**	**144.4**	**85.0**	**97.0**	**121.4**	**127.9**
	Straightforward	107.2	108.2	45.4	51.6	69.2	72.4
	Detailed	106.7	108.6	49.5	56.7	77.1	81.2
LLaVA v1.5-7B	Instruction	**114.5**	**122.7**	**83.9**	**99.2**	**106.3**	**117.5**
	Straightforward	60.3	65.1	23.4	28.0	36.1	40.4
	Detailed	40.6	44.1	17.6	21.1	26.2	29.4
Idefics2-8B (Base)	Instruction	**81.2**	**94.6**	**63.0**	**79.7**	**81.0**	**95.3**
	Straightforward	21.9	35.6	21.2	32.6	19.8	31.4
	Detailed	11.2	33.2	13.5	29.4	15.1	37.6
Idefics2-8B (Instruction-Tuned)	Instruction	83.6	90.1	62.3	74.7	84.3	93.0
	Straightforward	44.1	57.1	37.0	51.1	55.2	68.0
	Detailed	**91.8**	**99.6**	**72.5**	**85.5**	**91.6**	**101.0**
OpenFlamingo-9B	Instruction	**71.0**	**82.0**	**56.2**	**69.8**	**67.4**	**81.5**
	Straightforward	1.2	18.4	1.8	10.6	1.3	13.9
	Detailed	66.3	70.6	40.7	50.3	56.4	65.2

(1) Base detailed prompt:
I am an intelligent image captioning bot. Here are the features extracted by Q-Former for similar images along with their captions, following the format: [visual query tokens for example 1] [caption of example 1] ... [visual query tokens for example K]. Example 1: **<image$_1$>** caption 1, . . . , Example $K-1$: **<image$_{K-1}$>** caption $K-1$, **<image>** A creative short caption I can generate to describe example K is:

(2) Removing the initial phrase:
~~I am an intelligent image captioning bot.~~ Here are the features extracted by Q-Former for similar images along with their captions, following the format: [visual query tokens for example 1] [caption of example 1] ... [visual query tokens for example K]. Example 1: **<image$_1$>** caption 1, . . . , Example $K-1$: **<image$_{K-1}$>** caption $K-1$, **<image>** A creative short caption I can generate to describe example K is:

(3) Removing the word "creative":
I am an intelligent image captioning bot. Here are the features extracted by Q-Former for similar images along with their captions, following the format: [visual query tokens for example 1] [caption of example 1] ... [visual query tokens for example K]. Example 1: **<image$_1$>** caption 1, . . . , Example $K-1$: **<image$_{K-1}$>** caption $K-1$, **<image>** A ~~creative~~ short caption I can generate to describe example K is:

(4) Removing the initial phrase and the word "creative":
~~I am an intelligent image captioning bot.~~ Here are the features extracted by Q-Former for similar images along with their captions, following the format: [visual query tokens for example 1] [caption of example 1] ... [visual query tokens for example K]. Example 1: **<image$_1$>** caption 1, . . . , Example $K-1$: **<image$_{K-1}$>** caption $K-1$, **<image>** A ~~creative~~ short caption I can generate to describe example K is:

(5) Removing the name Q-Former:
I am an intelligent image captioning bot. Here are the features extracted ~~by Q-Former~~ for similar images along with their captions, following the format: [visual query tokens for example 1] [caption of example 1] ... [visual query tokens for example K]. Example 1: **<image$_1$>** caption 1, . . . , Example $K-1$: **<image$_{K-1}$>** caption $K-1$, **<image>** A creative short caption I can generate to describe example K is:

(6) Replacing Q-Former with a generic name:
I am an intelligent image captioning bot. Here are the features extracted by ~~Q-Former~~ **a model** for similar images along with their captions, following the format: [visual query tokens for example 1] [caption of example 1] ... [visual query tokens for example K]. Example 1: **<image$_1$>** caption 1, . . . , Example $K-1$: **<image$_{K-1}$>** caption $K-1$, **<image>** A creative short caption I can generate to describe example K is:

(7) Replacing Q-Former with a random name:
I am an intelligent image captioning bot. Here are the features extracted by ~~Q-Former~~ **XXXX** for similar images along with their captions, following the format: [visual query tokens for example 1] [caption of example 1] ... [visual query tokens for example K]. Example 1: **<image$_1$>** caption 1, . . . , Example $K-1$: **<image$_{K-1}$>** caption $K-1$, **<image>** A creative short caption I can generate to describe example K is:

Figure A1: Detailed prompts to evaluate the in-context learning capabilities of models in the image captioning task.

Table A3: **Influence of the number of demonstration examples on image captioning performance**. Evaluation results for image captioning task with the detailed prompt varying the number of demonstration examples (Shot) in the context. Bold numbers highlight the best performance for each model. We use the MS COCO training set as the knowledge base.

Model	Shot	MS COCO		Flickr30K		NoCaps	
		CIDEr-D (↑)	CIDEr-R (↑)	CIDEr-D (↑)	CIDEr-R (↑)	CIDEr-D (↑)	CIDEr-R (↑)
InstructBLIP Vicuna-7B (Q-Former fed with Caption)	Instruction	**147.2**	**149.0**	**86.3**	**98.2**	**124.2**	**130.5**
	0	136.2	139.1	80.5	92.7	116.1	123.2
	1	97.1	100.1	48.5	55.7	73.7	78.3
	3	94.4	97.9	42.8	50.3	68.9	74.3
	5	92.9	96.6	38.9	46.3	66.9	72.7
InstructBLIP Vicuna-7B (Q-Former fed with Instruction)	Instruction	**147.2**	**149.0**	**86.3**	**98.2**	**124.2**	**130.5**
	0	136.2	139.1	80.5	92.7	116.1	123.2
	1	110.6	112.7	52.6	59.8	81.5	85.4
	3	105.1	107.5	46.1	52.3	75.4	79.0
	5	102.4	105.2	44.7	50.3	73.5	77.1
InstructBLIP FlanT5-XL (Q-Former fed with Caption)	Instruction	**142.4**	**144.4**	**85.0**	**97.0**	**121.4**	**127.9**
	0	127.9	131.2	78.9	90.9	114.5	122.0
	1	54.9	56.3	21.9	25.1	44.4	47.1
	3	57.4	59.1	28.3	32.5	54.1	57.5
	5	72.3	74.8	28.1	32.3	54.3	57.5
InstructBLIP FlanT5-XL (Q-Former fed with Instruction)	Instruction	**142.4**	**144.4**	**85.0**	**97.0**	**121.4**	**127.9**
	0	127.9	131.2	78.9	90.9	114.5	122.0
	1	93.3	95.3	46.0	52.5	71.5	75.5
	3	106.7	108.6	49.5	56.7	77.1	81.2
	5	104.9	106.6	46.2	53.0	72.5	76.2
LLaVA v1.5-7B	Instruction	**114.5**	**122.7**	**83.9**	**99.2**	**106.3**	**117.5**
	0	82.1	87.2	55.4	65.4	77.2	83.9
	1	42.9	45.8	18.2	21.5	31.2	34.4
	3	40.3	43.8	17.5	21.3	25.2	28.1
	5	52.9	56.6	19.6	23.2	30.7	33.9
Idefics2-8B (Base)	Instruction	**81.2**	**94.6**	**63.0**	**79.7**	**81.0**	**95.3**
	0	11.0	36.9	16.1	36.8	13.7	37.0
	1	9.4	31.8	13.4	30.2	12.1	32.6
	3	11.2	33.2	13.5	29.4	15.1	37.6
	5	17.2	38.1	17.3	33.3	17.0	36.9
Idefics2-8B (Instruction-Tuned)	Instruction	83.6	90.1	62.3	74.7	84.3	93.0
	0	**102.4**	**110.4**	**76.5**	**91.5**	**99.7**	**110.1**
	1	94.1	102.9	72.7	88.1	93.6	104.4
	3	91.8	99.6	72.5	85.5	91.6	101.0
	5	88.7	96.1	70.3	82.6	88.1	97.0
OpenFlamingo-9B	Instruction	**71.0**	**82.0**	**56.2**	**69.8**	**67.4**	**81.5**
	0	52.1	66.0	38.8	49.4	43.5	61.4
	1	61.3	65.1	40.1	48.2	53.4	61.6
	3	66.3	70.6	40.7	50.3	56.4	65.2
	5	62.6	66.1	38.6	46.2	50.7	56.8

One can observe that, in the instruction-only scenario, Idefics2 and OpenFlamingo exhibit lower efficiency compared to InstructBLIP and LLaVA models. However, in the in-context learning setting, instruction-tuned Idefics2 follows the same scaling trend as InstructBLIP and LLaVA on Flickr30K and NoCaps. This indicates that Idefics2 (IT) benefits from additional contextual information as efficiently as InstructBLIP models and LLaVA with respect to the training data volume.

In contrast, OpenFlamingo consistently underperforms across all datasets. This finding aligns with Qin et al. (2024) and suggests that the fully autoregressive approach – where visual information is passed as input soft tokens to the LLM – is more data-efficient than OpenFlamingo's strategy of integrating visual information directly within the LLM's layers.

A.5 Qualitative Analysis

We qualitatively analyze how models' outputs vary between instruction-only scenarios and those with context, using both similar and random demonstrations. Specifically, we investigate whether the models effectively leverage contextual information. To do so, we select four examples from Flickr30K, as well as the demonstrations for these examples, as shown in Figure A3.

Consistent with our quantitative analysis presented in Section 4, InstructBLIP models can generate captions correctly describing the visual con-

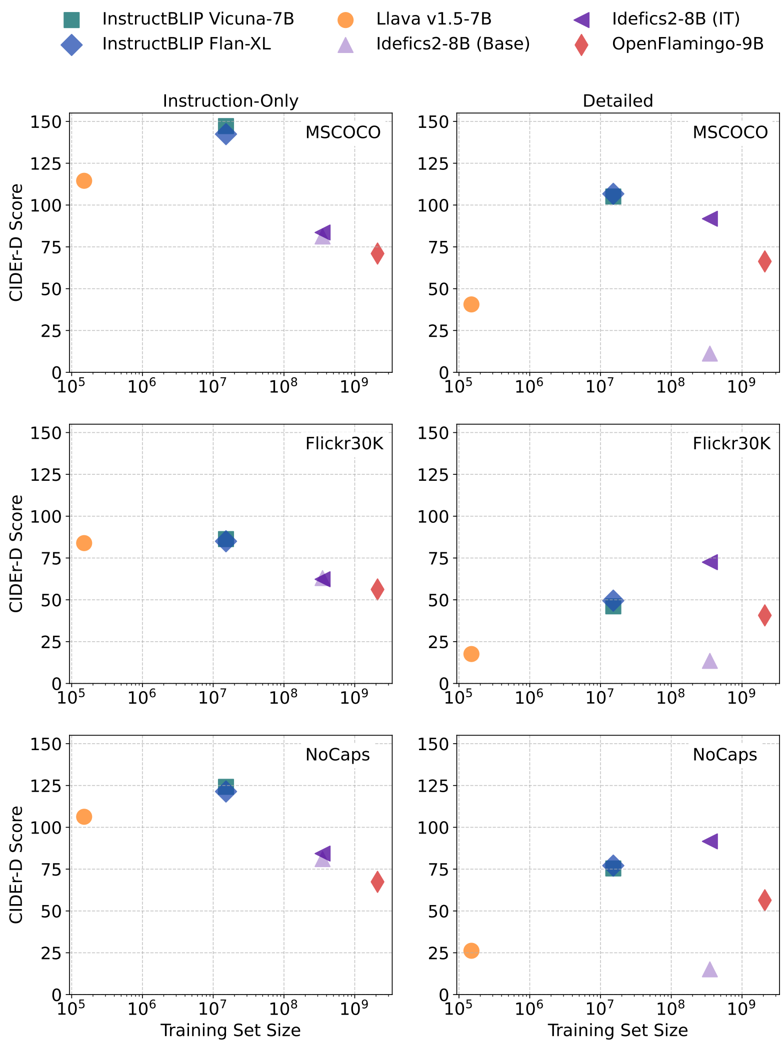

Figure A2: Influence of training dataset size on performance on instruction-only and in-context learning scenarios. Note that the training set size is in the log scale.

Table A4: **Influence of the number of demonstration examples on VQA performance**. Evaluation results for the VQA task varying the number of demonstration examples (Shot) in the context. We use the corresponding training set of each dataset as the knowledge base. We report the VQA accuracy, bold numbers highlight the best performance for each model.

Model	Shot	VizWiz (↑)	GQA (↑)	TextVQA (↑)	OKVQA (↑)
InstructBLIP Vicuna-7B	0	**21.8**	**49.1**	**33.8**	**42.4**
	1	19.8	47.6	29.6	41.2
	3	19.5	46.3	25.2	39.4
	5	19.5	45.2	23.5	38.2
InstructBLIP FlanT5-XL	0	21.0	**48.1**	**31.1**	**35.6**
	1	32.0	46.4	28.3	35.2
	3	**34.0**	45.1	26.2	34.7
	5	32.4	43.4	24.4	34.4
LLaVA v1.5	0	15.5	**56.3**	**37.7**	27.8
	1	15.2	45.5	10.7	27.1
	3	**20.7**	45.4	9.6	28.6
	5	19.9	44.9	10.3	**28.9**
Idefics2-8B (Base)	0	13.08	26.20	35.94	13.98
	1	26.05	18.49	26.34	12.33
	3	28.11	14.91	20.2	6.98
	5	32.98	19.38	21.1	7.6
Idefics2-8B (Instruction-Tuned)	0	23.3	**51.5**	**62.6**	**38.6**
	1	24.4	49.1	56.6	37.8
	3	26.3	49.9	57.2	38.4
	5	**27.1**	46.8	53.1	36.9
OpenFlamingo-9B	0	13.9	22.5	20.0	10.2
	1	29.0	24.9	19.7	14.9
	3	**33.8**	**29.1**	**22.8**	**17.5**
	5	20.9	23.1	16.8	12.3

tent of the query image in instruction-only and in-context learning with similar demonstrations. However, random demonstrations can confound the models, leading to captions unrelated to query images. Then, it confirms the performance gain shown in Figure 5. Furthermore, LLaVA proves to leverage visual cues provided within the demonstrations, often utilizing information from the first demonstration. Particularly, we can observe in example #1 that it incorporates the visual detail of the number 4 from the player's jersey from the last similar demonstration. And, in examples #2 and #4, it ignores the query image and describes exactly the first similar demonstration. In contrast, when random demonstrations are used, the model generates captions unrelated to the query image.

Idefics2 (IT) appears to ignore contextual information during caption generation. Interestingly, when given demonstrations, the model generates shorter captions that align with the average sentence length found in the MS COCO dataset. This behavior suggests that the model is attempting to replicate the structure of the provided examples rather than leveraging their semantic content. On the other hand, Idefics2 (Base) struggles with caption generation under the instruction-only scenario, often returning the prompt or part of it or even generating a text totally unrelated to the query image as in example #2. This indicates a reliance on contextual grounding. Nevertheless, its performance improves when given demonstrations, generating outputs similar to those of Idefics2 (IT), albeit with repetition of prompt segments at the end. This repetition suggests difficulty in processing instructions effectively. However, after instruction tuning, the model improves its handling of prompts and generates cleaner captions without extraneous prompt segments.

OpenFlamingo can generate captions in both instruction-only and in-context learning settings. However, when given demonstrations, it sometimes mixes up elements from different examples, resulting in captions unrelated to the query image. This is particularly seen in examples #2 and #3, where random demonstrations lead to captions incorporating unrelated elements. This suggests that while OpenFlamingo can utilize contextual information, it lacks a robust mechanism to differentiate between relevant and irrelevant context, compromising caption generation.

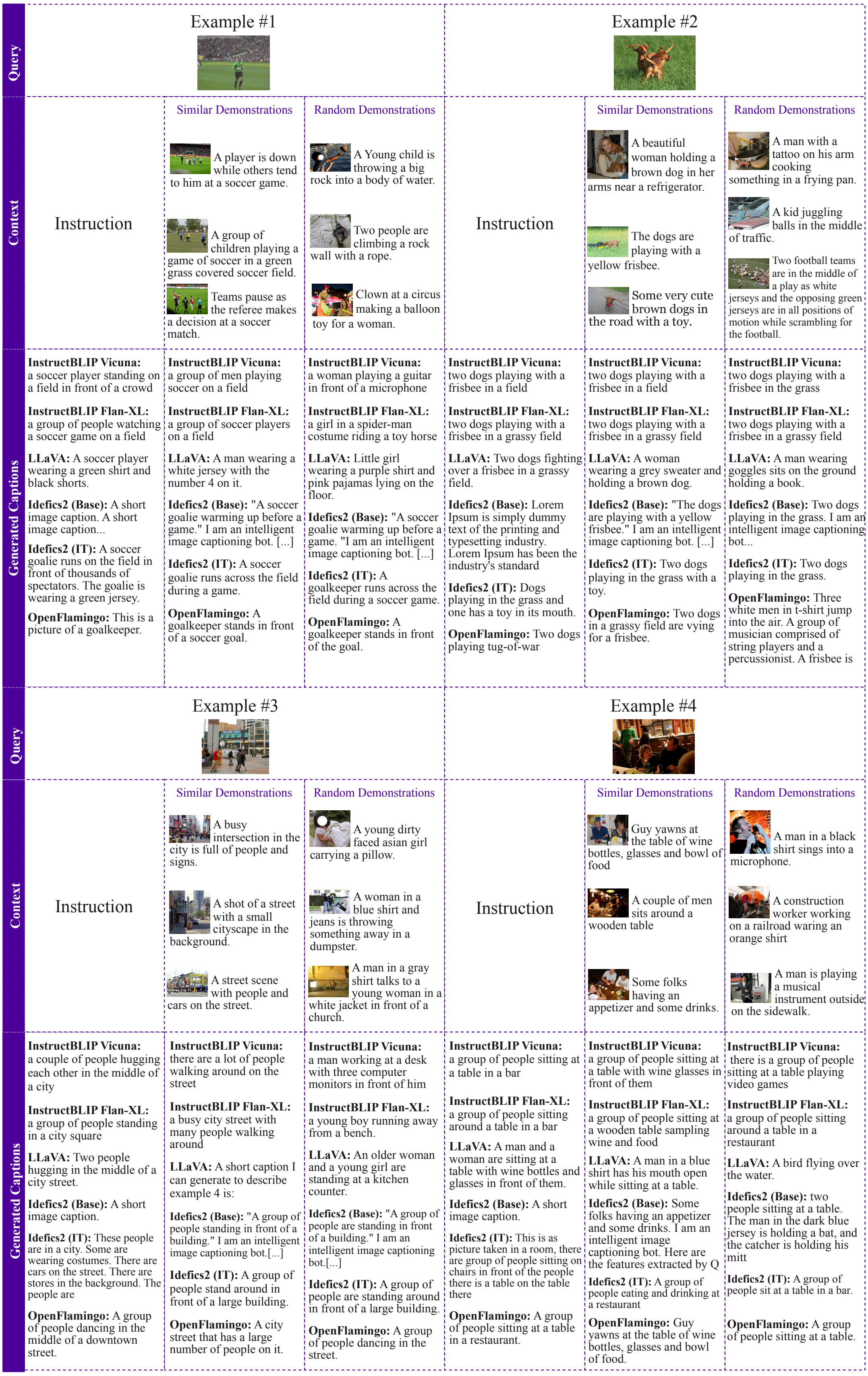

Figure A3: Selected examples from Flickr30K illustrate how models' outputs vary between instruction-only scenarios and those with context, using similar and random demonstrations. The demonstration examples are retrieved from the MS COCO training set, consistent with our experimental pipeline.

Table A5: **Ablation on detailed prompt**. Evaluation results for image captioning task under in-context learning testing detailed prompts with minor changes. Each sample includes three demonstration examples as context, with the MS COCO training set serving as the knowledge base. Instruction refers to the best performance across instruction-only prompts. Bold numbers highlight the best performance for each model.

Model	Prompt	MS COCO		Flickr30K		NoCaps	
		CIDEr-D (↑)	CIDEr-R (↑)	CIDEr-D (↑)	CIDEr-R (↑)	CIDEr-D (↑)	CIDEr-R (↑)
InstructBLIP Vicuna-7B	(1) Base detailed prompt	101.6	104.2	44.3	51.8	72.1	77.0
	(2) Removing the initial phrase	98.5	101.4	42.6	50.4	70.3	75.2
	(3) Removing the word "creative"	**106.2**	**108.5**	**46.8**	**53.0**	**73.6**	**77.2**
	(4) Removing both the initial phrase and the word "creative"	103.1	105.9	46.2	52.5	73.0	76.8
	(5) Removing the name "q-former"	101.1	103.6	42.6	49.8	70.6	75.4
	(6) Replacing "q-former" with a generic name	101.4	103.9	43.7	50.8	70.7	75.5
	(7) Replacing "q-former" with a random name	100.4	102.9	43.2	50.0	71.1	76.0
InstructBLIP FlanT5-XL	(1) Base detailed prompt	0.4	0.4	0.0	0.0	0.5	0.6
	(2) Removing the initial phrase	0.7	0.7	0.1	0.1	2.0	2.2
	(3) Removing the word "creative"	**106.7**	**108.6**	**49.6**	**56.7**	**76.1**	**80.2**
	(4) Removing both the initial phrase and the word "creative"	105.0	106.8	48.6	55.7	75.3	79.3
	(5) Removing the name "q-former"	0.5	0.5	0.0	0.1	1.1	1.2
	(6) (6) Replacing "q-former" with a generic name	0.2	0.2	0.0	0.1	0.4	0.4
	(7) Replacing "q-former" with a random name	0.3	0.3	0.0	0.1	0.2	0.3
LLaVA v1.5	**(1) Base detailed prompt**	38.9	43.5	**18.3**	**22.2**	**26.6**	**30.5**
	(2) Removing the initial phrase	29.3	32.0	12.8	15.4	18.4	20.7
	(3) Removing the word "creative"	**40.6**	44.1	17.6	21.1	26.2	29.4
	(4) Removing both the initial phrase and the word "creative"	29.0	31.4	12.9	15.6	17.0	19.3
	(5) Removing the name "q-former"	39.5	**44.5**	17.2	21.0	26.0	29.5
	(6) Replacing "q-former" with a generic name	39.4	44.0	17.4	21.3	25.9	29.7
	(7) Replacing "q-former" with a random name	39.8	44.4	17.8	21.6	26.3	29.9
Idefics2-8B	(1) Base detailed prompt	66.5	74.4	53.0	64.2	65.6	74.0
	(2) Removing the initial phrase	69.5	76.7	54.0	64.5	69.7	77.8
	(3) Removing the word "creative"	**91.8**	**99.6**	**72.5**	**85.5**	**91.6**	**101.0**
	(4) Removing both the initial phrase and the word "creative"	87.7	95.4	66.9	80.3	88.4	98.8
	(5) Removing the name "q-former"	65.0	72.7	51.0	61.6	64.0	72.3
	(6) Replacing "q-former" with a generic name	64.0	72.3	51.5	62.6	62.5	70.9
	(7) Replacing "q-former" with a random name	67.6	75.9	54.4	65.4	67.3	76.0
OpenFlamingo-9B	(1) Base detailed prompt	66.9	72.5	43.9	53.8	57.5	66.9
	(2) Removing the initial phrase	**67.2**	**72.9**	**44.4**	**53.8**	**58.8**	**68.0**
	(3) Removing the word "creative"	66.3	70.6	40.7	50.3	56.4	65.2
	(4) Removing both the initial phrase and the word "creative"	66.2	70.8	43.2	52.5	58.3	66.9
	(5) Removing the name "q-former"	65.6	71.4	43.0	52.5	56.4	66.0
	(6) Replacing "q-former" with a generic name	66.3	72.1	44.4	54.1	56.9	67.1
	(7) Replacing "q-former" with a random name	66.5	72.3	42.7	51.8	56.6	65.9

A.5.1 VizWiz

To investigate why adding in-context demonstrations benefits performance on the VizWiz dataset, unlike other datasets, we first analyze the distribution of answers in this dataset. Figure A4 shows the answers that appear more than 10 times in the dataset. A clear imbalance can be seen, with "unanswerable" being by far the most frequent answer. It seems approximately 1,750 times, whereas all other answers occur fewer than 100 times, with most appearing only once. We hypothesize that this imbalance explains the performance gains observed when the number of demonstrations increases. To test this, we examine the top 10 most frequently generated answers from each model in the 0-shot setting, where only the question is provided, and the 5-shot setting, as shown in Figure A5.

In the 0-shot setting, the distribution of answers is more balanced, and in most cases, "unanswerable" does not even appear in the top 10. However, when demonstrations are included, "unanswerable" becomes the most frequent answer, with a significant margin over others for most models. Notably, InstructBLIP FlanT5-XL, Idefics2 (Base), and OpenFlamingo generate "unanswerable" most often, with InstructBLIP FlanT5-XL and OpenFlamingo outputting it incorrectly many times. This suggests that demonstrations labeled as "unanswerable" influence the models to replicate this response, leading to improved performance due to the dataset's strong bias toward this answer.

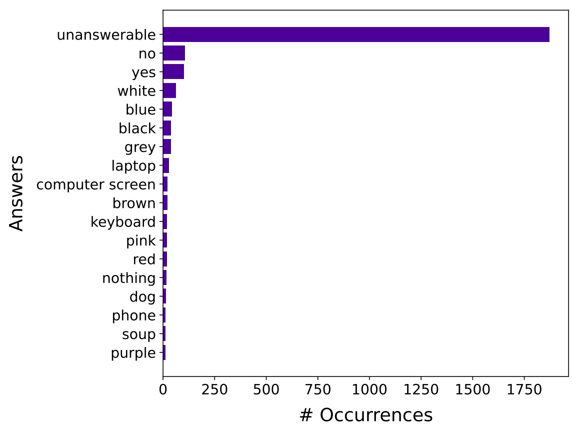

Figure A4: Answers that occur more than 10 times in the VizWiz dataset.

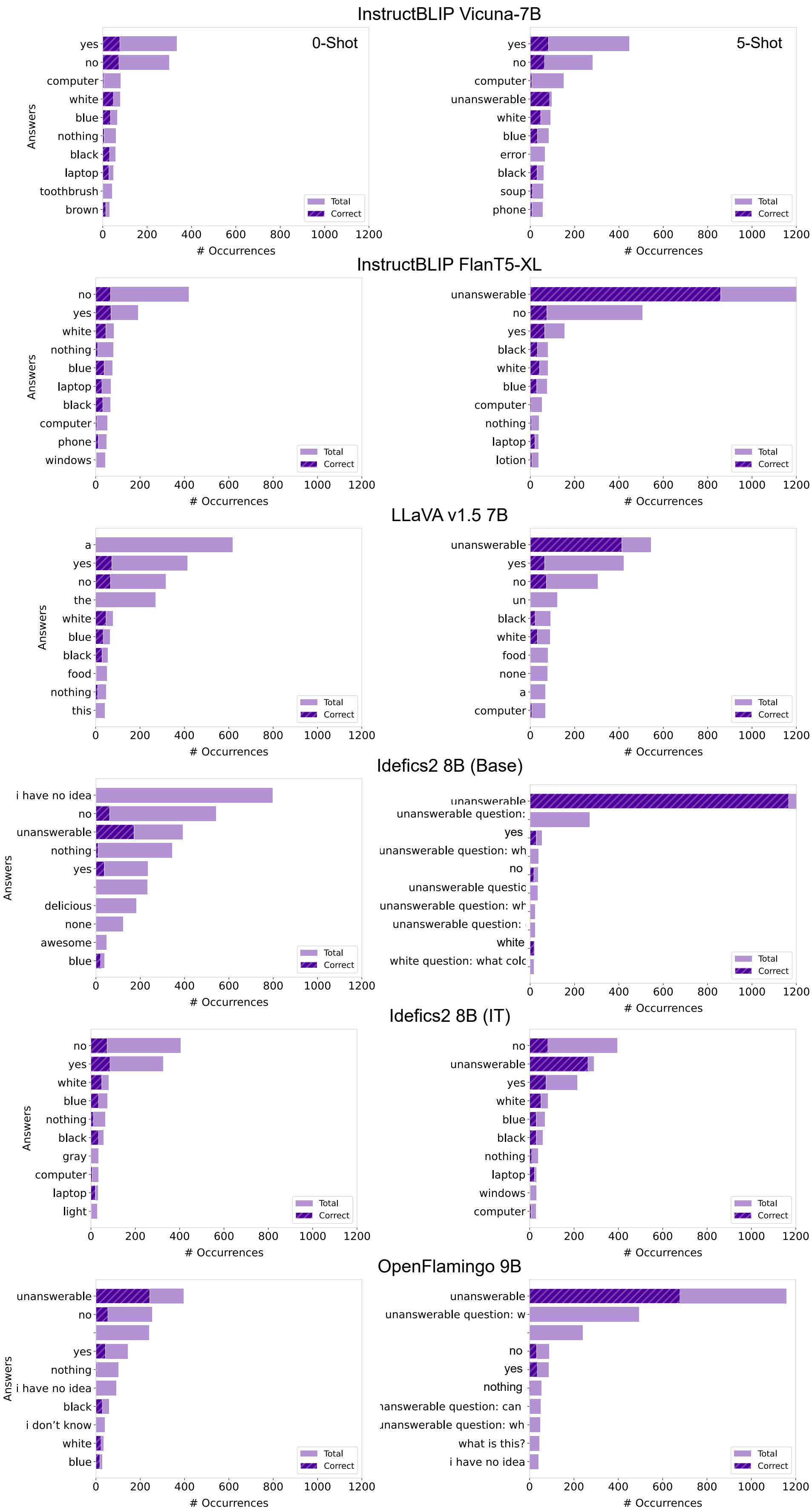

Figure A5: Top 10 most frequent answers for each model on the VizWiz dataset in both 0-shot and 5-shot scenarios. "Total" refers to the total number of occurrences of a given answer and "Correct" indicates the number of correct ones.

A.6 More Details on Experimental Setup

To facilitate the reproducibility of our work, we report in Table A6 the models we analyzed, along with details on their number of parameters and training set size. Table A7 shows the datasets used in our experiments, including basic statistics on their size. Additionally, we outline the main hyperparameters used in our experiments in Tables A8 and A9. Table A8 lists the hyperparameters specific to image captioning, while Table A9 includes those used for VQA. We conducted our experiments in a heterogeneous computing environment; however, the majority were performed on a single Quadro RTX 8000 GPU. Also, all experiments were conducted only once.

Table A6: VLMs investigated in this work. For each model, we report the number of parameters and the size of the dataset used for training.

Model	#Params (B)	Training set size (M)
Llava v1.5-7B	7.1	0.15
InstructBLIP Vicuna-7B	7.9	15.1
InstructBLIP Flan-XL	4.0	15.1
Idefics2-8B	8.4	351.2
OpenFlamingo-9B	8.1	2,101.0

Table A7: Datasets used in our experiments. For each dataset, we report the number of samples in each split and the specific task it is used for. Note that we do not use Flickr or NoCaps training sets, as we rely on the MS COCO training set as the knowledge base for these datasets. "Val." stands for the validation dataset.

Dataset	Size	Task
MS COCO	Train: 118.2K/ Val: 5.0K	Image Captioning
Flickr30K	Val: 1.0K	Image Captioning
NoCaps	Val: 4.5K	Image Captioning
VizWiz	Train: 20.5K/ Val: 4.3K	VQA
GQA	Train: 943K/ Val: 12.5K	VQA
TextVQA	Train: 34.6K/ Val: 5K	VQA
OKVQA	Train: 9K/ Val: 5K	VQA

Table A8: Hyperparameters for image captioning.

Hyperparameters	Value
# Beams	5
Max. New Tokens	30
Min. Length	10
Repetition Penalty	1.5
Length Penalty	1.0
Temperature	1.0

Table A9: Hyperparameters for VQA.

Hyperparameters	Value
# Beams	5
Max. New Tokens	10
Min. Length	1
Length Penalty	-1.0