Parameter-Efficient Fine-tuning of InstructBLIP for Visual Reasoning Tasks

Sungkyung Kim[1*] **Adam Lee**[2*] **Junyoung Park**[1]
Sounho Chung[1] **Jusang Oh**[1] **Jay-Yoon Lee**[3†]
[1]Seoul National University [2]UC Berkeley
[3]Graduate School of Data Science, Seoul National University
{sk0428, jyp0314, aschung01, dhwntkd412, lee.jayyoon}@snu.ac.kr
alee00@berkeley.edu

Abstract

Visual language models have recently demonstrated enhanced capabilities in visual reasoning tasks by employing external modules upon language models for visual language alignment. InstructBLIP uses a Q-Former and a projection layer to convert input image embeddings into soft visual prompts to enhance the instruction-following capabilities of large language models (LLMs). Although fine-tuning InstructBLIP has shown great results in downstream tasks, previous works have been restrictive, only full fine-tuning the Q-Former, while freezing the LLM. In this work, we investigate the performance of the PEFT method, LoRA, on both the Q-Former and the base LLMs, specifically Flan-T5-XL and Vicuna-7B, using visual reasoning benchmarks ScienceQA and IconQA. We observe that, when the LLM is frozen, training the Q-Former with LoRA achieves comparable performance to full fine-tuning using under 2% of the trainable parameters. Furthermore, fine-tuning the LLM consistently result in better performances than InstructBLIP. Lastly, applying LoRA to both the LLM and the Q-Former surpasses the performance of only full fine-tuning the Q-Former while using less than 12% of the trainable parameters. These results highlight the effectiveness of applying PEFT to visual language models for visual reasoning tasks. The code is available at `https://github.com/AttentionX/InstructBLIP_PEFT`.

1 Introduction

Pre-trained large language models can be fine-tuned to achieve high performance on many tasks. For instance, instruction tuning has been proposed to align the model's responses more closely with human intentions [1, 2]. This approach is also applicable in multi-modal settings with visual instruction tuning, that enhances the model's capabilities to follow instructions for visual question answering and visual reasoning tasks. LLaVA [3, 4] uses a projection layer to convert the CLIP [5] image embeddings to the word embedding space of language models, and trains both the projection layer and the language model. BLIP-2 [6] and InstructBLIP [7] use a Q-Former for visual-language alignment, similarly to Perceiver IO [8], which extract visual features in a fixed number of learnable embeddings (32 in BLIP-2). InstructBLIP only fine-tunes the Q-Former and a fully connected projection layer while freezing the LLM. The Q-Former is especially significant for its role aligning several modalities with cross-attention and encoding the information in a small number of learnable embeddings. This multimodal alignment approach has been adopted in several recent studies, including Video-LLaMA [9] and Qwen-VL [10].

[*]Equal contribution.
[†]Corresponding author.

Efficient Natural Language and Speech Processing (NeurIPS 2023).

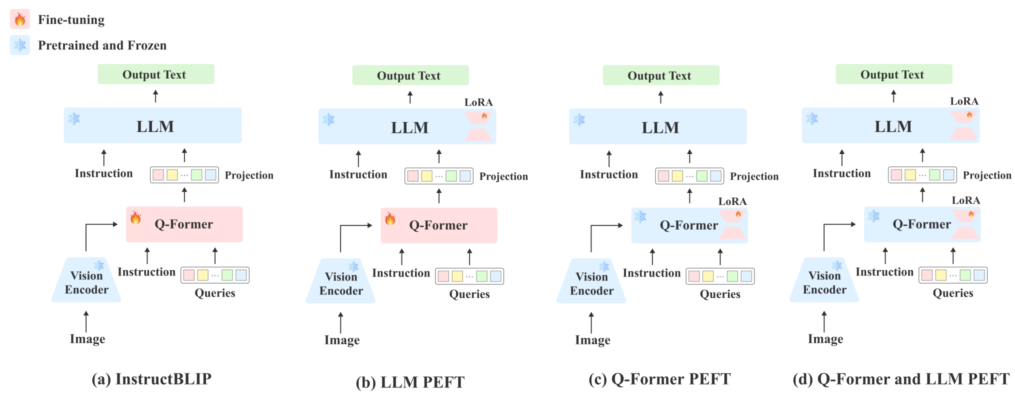


Figure 1: Applying PEFT to different components in InstructBLIP.

While InstructBLIP and LLaVA achieve competitive performance on several downstream visual reasoning benchmarks [3, 7], each visual language model has its own limitations. In the case of LLaVA, full fine-tuning the large language model may be costly, due to the computational memory required to update billions of parameters. For InstructBLIP, freezing the LLM model may hinder the model from learning task-specific language understanding and generation abilities. Parameter efficient fine-tuning (PEFT) can mitigate both problems by fine-tuning large models with much less computational memory while still maintaining competitive performance [11, 12, 13, 14, 15, 16, 17, 18]. Although various PEFT methods perform competitively on downstream tasks [13, 19, 20], the efficacy of PEFT methods, both for visual reasoning tasks and for visual-language alignment models like the Q-Former, remains under-explored.

In this work, we evaluate the performance of PEFT on InstructBLIP with two benchmarks, ScienceQA [21] and IconQA [22], that respectively test knowledge-grounded visual reasoning and abstract visual reasoning capabilities. Specifically we apply LoRA to the Q-Former and base LLMs, Flan-T5-XL [23] and Vicuna-7B [24], in InstructBLIP and test with 3 different settings: applying LoRA only to the LLM, applying LoRA only to the Q-Former, and applying LoRA to both the LLM and the Q-Former (Figure 1). We also comprehensively test the performance of LoRA applied to different sublayers in the transformer with different ranks. To the best of our knowledge, we are the first to inspect the effectiveness of PEFT methods on the Q-Former for visual reasoning.

Our contributions can be summarized as follows: (1) Our experiments reveal that applying PEFT on the LLM, rather than freezing, consistently results in better performance than InstructBLIP. (2) We demonstrate that applying PEFT to the Q-Former reduces trainable parameters to less than 2% while maintaining comparable performances. (3) We show that, in contrast to full fine-tuning the Q-Former and freezing the LLM, applying PEFT on both components can achieve superior results and bring down the total trainable parameters to less than 12%.

2 Method

In this work, we apply the PEFT method LoRA [11] to two different components in InstructBLIP, the Q-Former and the LLM, and evaluate the performance on two visual reasoning benchmarks, ScienceQA and IconQA.

LoRA greatly reduces trainable parameters by decomposing weight update matrix into the product of two low rank matrices $\Delta W = BA$. After the fine-tuning, weight matrix can be reparametrized by adding the weight update to the original pre-trained model weights: $W + \Delta W = W + BA$, where $W \in \mathbb{R}^{d\times k}$, $B \in \mathbb{R}^{d\times r}$, $A \in \mathbb{R}^{r\times k}$, $r \ll \min(d, k)$. This process can prevent additional latency during inference. Unlike the original LoRA implementation, which confines its application to only the self-attention modules [11], our approach extends the use of LoRA to multiple transformer sublayers in both the Q-Former and the LLM. For the Q-Former, we apply LoRA to the query and value projection layers in the self-attention layers and to the query, key, value, and output projection layers with the cross-attention layers. We also apply LoRA to the feed-forward networks. Conversely,

Method				ScienceQA				IconQA			
LLM	Q-Former	Sublayer	Base Model	r=1	r=2	r=4	r=8	r=1	r=2	r=4	r=8
LoRA	Full	ffn	Flan-T5-XL	**86.42**	86.37	85.32	86.27	73.40	**74.76**	74.00	71.52
LoRA	Full	attn	Flan-T5-XL	**87.36**	86.17	86.91	86.42	72.34	72.88	**73.45**	73.29
LoRA	Full	all	Flan-T5-XL	87.41	87.36	**<u>88.20</u>**	87.90	72.61	**<u>75.06</u>**	73.23	72.74
Freeze	LoRA	ffn	Flan-T5-XL	84.83	83.79	83.14	**85.87**	70.54	72.13	68.08	**72.40**
Freeze	LoRA	self-attn	Flan-T5-XL	**86.02**	83.74	79.57	86.02	71.82	**72.55**	72.06	71.64
Freeze	LoRA	cross-attn	Flan-T5-XL	84.13	**86.32**	84.88	85.18	72.32	72.42	72.32	**<u>73.92</u>**
Freeze	LoRA	all	Flan-T5-XL	85.37	86.42	83.89	**<u>86.61</u>**	70.19	70.50	72.82	**73.31**
LoRA	LoRA	all	Flan-T5-XL	88.00	88.10	**<u>88.35</u>**	88.05	71.47	**<u>73.34</u>**	71.41	73.18
LoRA	Full	ffn	Vicuna-7B	86.32	**86.42**	85.87	85.97	71.39	72.97	**73.02**	72.34
LoRA	Full	attn	Vicuna-7B	**<u>86.42</u>**	86.32	85.08	85.23	72.36	**73.16**	72.29	73.02
LoRA	Full	all	Vicuna-7B	85.03	**86.32**	85.57	85.72	**<u>73.77</u>**	71.71	72.93	73.15
Freeze	LoRA	ffn	Vicuna-7B	83.44	**83.74**	83.64	83.74	69.89	**72.50**	72.50	71.11
Freeze	LoRA	self-attn	Vicuna-7B	**83.19**	81.51	82.25	83.14	71.23	71.45	71.42	**71.74**
Freeze	LoRA	cross-attn	Vicuna-7B	**83.29**	83.24	83.14	82.75	71.11	72.40	71.99	**73.39**
Freeze	LoRA	all	Vicuna-7B	**<u>85.18</u>**	82.80	83.74	83.44	71.49	**<u>73.92</u>**	71.45	73.40
LoRA	LoRA	all	Vicuna-7B	85.87	**<u>87.11</u>**	85.08	85.62	71.72	72.01	72.61	**<u>73.05</u>**

Table 1: Overall performance results. "Full" indicates full fine-tuning, and the best results among 4 r values are bolded. The best results for each PEFT category, benchmark, and base language models are underlined. The underlined performances are used to compare the best performances between PEFT methods in Figure 2.

in the LLM, we apply LoRA to both the query and value layers in the attention module and to the feed-forward network.

Base Models and Benchmarks. We selected InstructBLIP as the base model given its reported state-of-the-art performance for fine-tuning on several downstream tasks [7], including ScienceQA (IMG) [21], OCR-VQA [25], and A-OKVQA [26]. We use the InstructBLIP implementation of LAVIS [27] and use pre-trained Flan-T5-XL[1] and Vicuna-7B[2] HuggingFace checkpoints in our experiments.

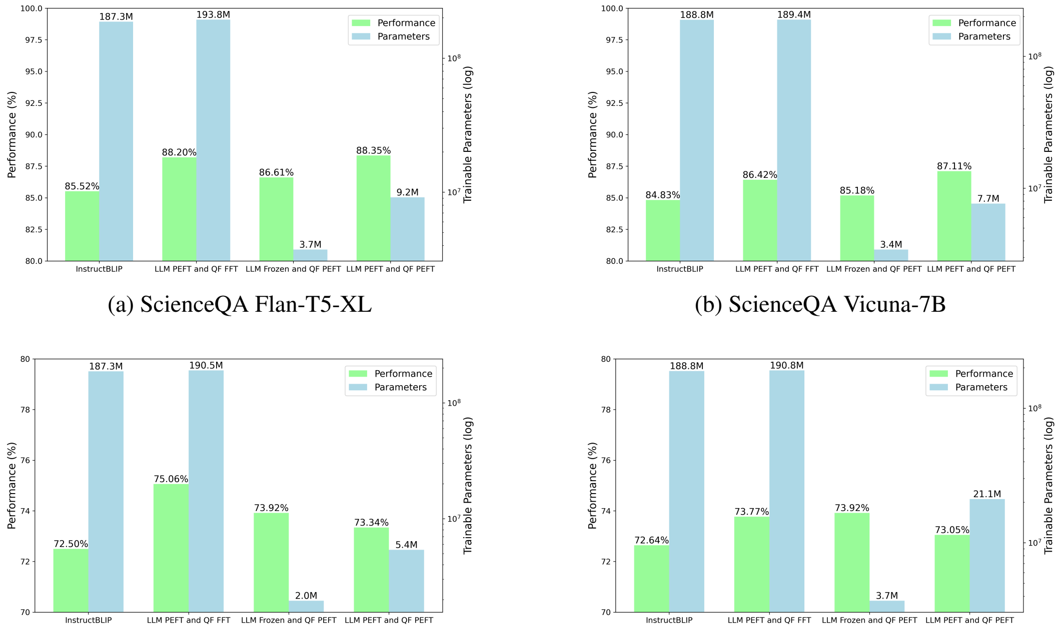
We use two benchmarks from InstructBLIP covering tasks of Knowledge Grounded Visual Reasoning (ScienceQA) [21] and Abstract Visual Reasoning (IconQA) [22]. These benchmarks were held-out datasets of InstructBLIP, and were not involved in training the baseline InstructBLIP model.

Knowledge Grounded Visual Reasoning is a task of answering questions with a provided image related to the knowledge in diverse academic areas including physics, biology, and math. We use the ScienceQA dataset which covers a variety of science topics with corresponding extensive explanations. We only use the questions with image context (IMG). ScienceQA (IMG) has 6.2k training samples and 2.1k, 2.0k samples for validation and testing.

Abstract Visual Reasoning is a task of answering questions after comprehending the abstract meanings from an image. We use IconQA which contains question-answer pairs for natural images that require comprehensive reasoning abilities to understand abstract diagrams.

Experimental setup. In the original InstructBLIP [7], full fine-tuning was applied to the Q-Former, while the LLM was frozen. In this work, we empirically analyze the effectiveness of training LoRA on the Q-Former and the LLM. (1) First, we apply LoRA to the LLM while still full fine-tuning the Q-Former, so the LLM is further trained to adapt to visual reasoning tasks. (2) Second, we apply LoRA to the Q-Former while freezing the LLM, resulting in efficient fine-tuning of the Q-Former. (3) Finally, we apply LoRA to both the Q-Former and the LLM. The evaluation entail testing with different ranks (1, 2, 4, 8), and for the base models of Flan-T5-XL and Vicuna-7B. The main results of the overall experiments are in Table 1, and the performance comparison of (1), (2), (3) and the original InstructBLIP is in Figure 2. The implementation and training details can be found in Appendix A, and instruction templates used for instruction tuning can be found in Appendix B.

[1] `https://huggingface.co/google/flan-t5-xl`
[2] `https://huggingface.co/lmsys/vicuna-7b-v1.3`



(a) ScienceQA Flan-T5-XL

(b) ScienceQA Vicuna-7B

(c) IconQA Flan-T5-XL

(d) IconQA Vicuna-7B

Figure 2: Performance and trainable parameter comparison among three PEFT methodologies using base models Flan-T5-XL and Vicuna-7B on ScienceQA and IconQA benchmarks. This compares the best performing configurations (rank value and LoRA-applied sublayers) between InstructBLIP (Q-Former full fine-tuning with frozen LLM), LLM PEFT with Q-Former full fine-tuning, Q-Former PEFT with frozen LLM, and Q-Former PEFT with LLM PEFT. "QF" denotes Q-Former. "FFT" denotes full fine-tuning.

3 Experiments

PEFT on LLM. We assess the efficacy of fine-tuning the LLM in InstructBLIP using LoRA. We consider 3 configurations: applying LoRA to the attention modules, the feed-forward network (FFN), and both the attention modules and the FFN. Across all tasks, with both Flan-T5-XL and Vicuna-7B as base models, fine-tuning the LLM with LoRA consistently outperforms InstructBLIP, as shown in Figure 2. These results suggest that introducing additional trainable parameters in the LLM enhances its language reasoning abilities for visual reasoning tasks. We find no clear performance differences among the LoRA ranks (1, 2, 4, 8). Also, unlike previous studies on language models [14, 28], no particular sublayer stood out in performance with LoRA.

PEFT on Q-Former. We examine the effectiveness of applying LoRA to different sublayers in the Q-Former while keeping the LLM frozen. This involves training LoRA on the self-attention, cross-attention, and FFN layers individually, and collectively on all three layers. We initially hypothesized that the cross-attention layer, given its direct role in image feature extraction, would be the most effective. However we observe no notable performance differences among the LoRA sublayer configurations. LoRA on Q-Former either outperform or match the results of full Q-Former fine-tuning while utilizing less than 2% of the trainable parameters (Figure 2). This suggests that training LoRA on the Q-Former offers significantly more efficient training while maintaining competitive performance. Furthermore, higher LoRA ranks do not result in better performance, indicating that the Q-Former's low-rank weight updates for learning visual reasoning only require small intrinsic ranks [11].

PEFT on both LLM and Q-Former. Finally we apply LoRA to both the Q-Former and the LLM, using the same rank for all possible sublayers in both components. Our results show that this approach outperforms InstructBLIP for both base LLMs across both benchmarks, using fewer than 12% of trainable parameters (as depicted in Figure 2). A notable observation is that the performance gap is higher in ScienceQA than in IconQA. This discrepancy can be attributed to ScienceQA's richer language context. Given that ScienceQA entails more language information than IconQA, training the language model appears to yield a greater boost in performance.

4 Conclusion

In this study, we systematically evaluate the benefits of applying LoRA to the Q-Former and LLM of InstructBLIP for visual reasoning tasks. Our results show that applying PEFT to the LLM leads to improved performance compared to InstructBLIP. Additionally, by employing PEFT on the Q-Former, we achieve outcomes comparable to full fine-tuning while only utilizing less than 2% of its parameters. Finally, we find that training both the LLM and Q-Former with PEFT yields superior results while training on less than 12% of the parameters compared to InstructBLIP. These findings hold practical importance; our results recommend jointly training both the Q-Former and LLM using PEFT, especially when computational resources are limited. Given our findings that demonstrate the efficiency and effectiveness of PEFT methods on InstructBLIP, we believe this work lays the foundation and motivate further research into efficient visual instruction tuning methods.

5 Acknowledgements

This work was supported by Seoul National University's Engineering Department and the NLP Research Group AttentionX. We are grateful for the support of A100 GPUs.

References

[1] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`, 2023.

[2] Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions, 2023.

[3] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023.

[4] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023.

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[6] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023.

[7] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023.

[8] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier Hénaff, Matthew M. Botvinick, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver io: A general architecture for structured inputs outputs, 2022.

[9] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding, 2023.

[10] Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023.

[11] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[12] Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou, et al. K-adapter: Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv:2002.01808*, 2020.

[13] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023.

[14] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366*, 2021.

[15] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019.

[16] Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks. *arXiv preprint arXiv:2110.07602*, 2021.

[17] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

[18] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*, 2021.

[19] Peng Gao, Jiaming Han, Renrui Zhang, Ziyi Lin, Shijie Geng, Aojun Zhou, Wei Zhang, Pan Lu, Conghui He, Xiangyu Yue, Hongsheng Li, and Yu Qiao. Llama-adapter v2: Parameter-efficient visual instruction model, 2023.

[20] Shamil Ayupov and Nadezhda Chirkova. Parameter-efficient finetuning of transformers for source code. *arXiv preprint arXiv:2212.05901*, 2022.

[21] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.

[22] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*, 2021.

[23] Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*, 2022.

[24] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.

[25] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 947–952, 2019.

[26] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-okvqa: A benchmark for visual question answering using world knowledge, 2022.

[27] Dongxu Li, Junnan Li, Hung Le, Guangsen Wang, Silvio Savarese, and Steven C.H. Hoi. LAVIS: A one-stop library for language-vision intelligence. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 31–41, Toronto, Canada, July 2023. Association for Computational Linguistics.

[28] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

A Model Training Details

We conduct each experiment in Table 1 and Figure 2 using a single A100 GPU. We set the maximum epoch to 15 with early stopping of 3 patience steps. We use linear decay as a learning rate scheduler with the AdamW optimizer. For the initial learning rate, we primarily use 2e-5 for experiments which involves full fine-tuning the Q-Former, and otherwise 5e-4. For certain cases, we lower the learning rate (from 2e-5 to 1e-5, and 5e-4 to 1e-4) for effective training. These cases include: (1) When the model is trained on less than 8 epochs (the halfway point) by early stopping, (2) When the training is considered unstable, i.e. resulting in over 10%p lower performance than other experiment in an equivalent setup having different r value. We set the weight decay to 0.05. For batch size, we use 16 as an effective batch size across all experiments. Only difference is that (batch size, gradient accumulation iterations) were set to (8, 2) for Vicuna-7B and (16, 1) for Flan-T5-XL.

B Instruction Templates

We provide instructions used in ScienceQA and IconQA. We use the same format from the InstructBLIP paper. We add alphabet labels for each choices and the answer. For ScienceQA, we construct the "context" section of the instruction by incorporating information from both the 'hint' and 'lecture' fields, if they are available in the dataset.

ScienceQA <Image> Context: { {hint} {lecture} } Question: { {question} } Options: { {choices} }. Answer:

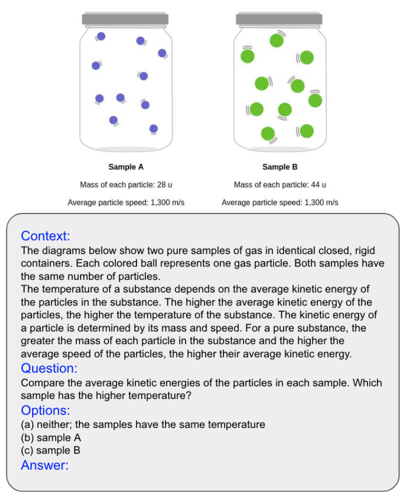


Figure 3: Example ScienceQA[3]instruction template.

IconQA <Image> Question: { {question} } Options: { {choices} }. Short answer:

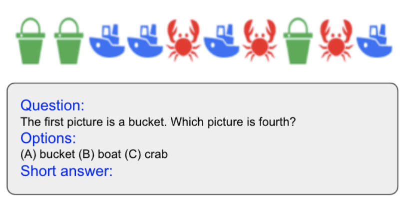


Figure 4: Example IconQA[3] instruction template.

[3]`https://creativecommons.org/licenses/by-nc-sa/4.0/`