MAP: Low-data Regime Multimodal Learning with Adapter-based Pre-training and Prompting

Wenyan Li *
University of Copenhagen
weli@di.ku.dk

Dong Li *
Shanghai AI Lab
lidong@ailab.org.cn

Wanjing Li *
Northeastern University (China)
Leawnn@163.com

Yuanjie Wang *
Beijing Institute of Technology
njwangyuanjie@163.com

Hai Jie *
Beihang University
jsea@buaa.edu.cn

Yiran Zhong *
Shanghai AI Lab
zhongyiran@ailab.org.cn

Abstract

Pretrained vision-language (VL) models have shown impressive results on various multimodal downstream tasks recently. Many of the benchmark models build on pretrained causal language models (LMs), leveraging the original few-shot learning and generalization capability of the LMs trained with large text corpora. However, these models are often gigantic and require large-scale image and text data with high computational cost to train. This paper introduces a moderate-size model called MAP for efficient VL transfer learning through adapter-based pretraining and prompting. We aim to answer the question of how much we can complete through VL pretraining within the low-data regime while maximizing efficiency in transferring knowledge of a moderate-size frozen LM. Our experiments demonstrate that MAP achieves substantially better zero-shot and few-shot performance on downstream VL tasks with only 10% the size of pretraining data and a 30× lighter pretrained LM backbone compared to Frozen. MAP also outperforms fully trained models of comparable size at retaining its transfer learning ability when the amount of training data reduces.

1 Introduction

Recent vision-language models commonly leverage pre-trained language models (LMs) on various multimodal tasks. It is crucial for them to retain the original generation capability of the LMs while efficiently incorporating the knowledge from new modalities. A line of work has shown impressive generalization and transfer ability of vision-language models that build on large causal decoder-only LMs (Alayrac et al., 2022; Tsimpoukelli et al., 2021; Eichenberg et al., 2022; Wang et al., 2021a). While powerful, these GPT-style pretrained LMs request high computing machines for deployment.

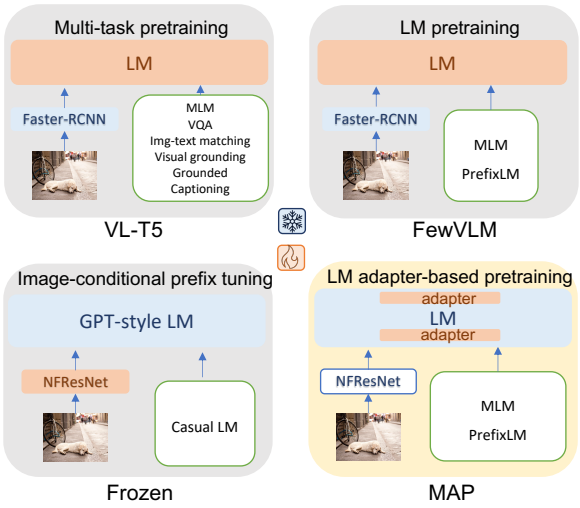

Figure 1: Comparison between VL learning with multitask pretraining, LM pretraining, image-conditional prefix tuning and adapter-based LM pretraining.

Compared to decoder-only LMs, Wang et al. (2021c) shows that encoder-decoder model introduces an inductive bias that decouples multimodal feature encoding from generation, yielding improved performance on downstream tasks. Recent research by Jin et al. (2021) also demonstrates that VL models pretrained with a moderate-size encoder-decoder LM backbone can be strong few-shot learners. However, in these approaches, the parameters of the language model were entirely updated while learning vision inputs (Cho et al., 2021; Jin et al., 2021; Wang et al., 2021c), or involving more task-specific data during multitask pretraining or fine-tuning (Sung et al., 2022; Cho et al., 2021).

Naturally, taking account of both model structure and pretraining efficiency, we improve on previous models and introduce an encoder-decoder parameter-efficient VL model, MAP. As shown in Figure 1, we apply adapters for VL pretraining with masked language modeling (Masked LM) and prefix language modeling (PrefixLM) objectives. We keep the backbone encoder-decoder language

*Work done while employed at SenseTime

Proceedings of the 2023 CLASP Conference on Learning with Small Data, pages 185–190
September 11–12, 2023.

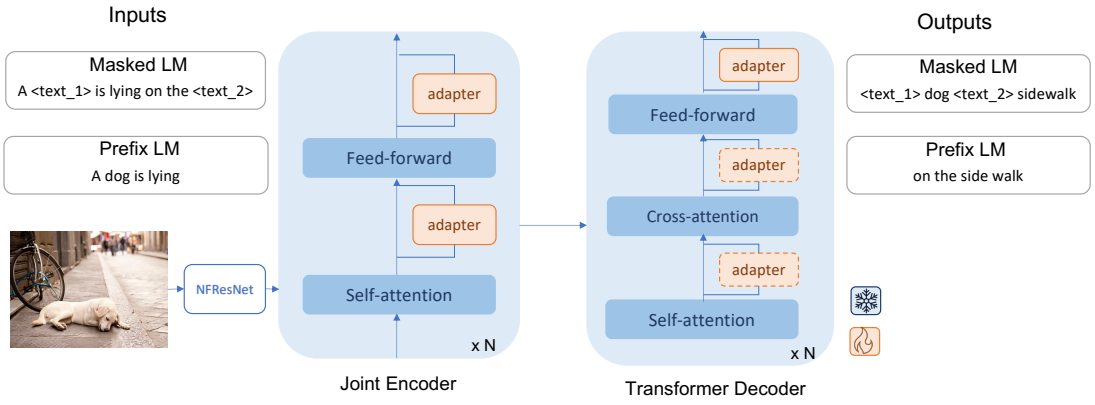

Figure 2: Illustration of the VL pretraining process and the model structure of MAP. We experiment with settings on pretraining by only updating adapters or updating both adapters and the NFResNet image encoder. In the decoder, we experiment with settings of adding adapters to self-attention and cross-attention.

model frozen. In downstream tasks, we provide task-specific prompts to guide the pretrained model for few-shot learning. Our moderate-size model substantially outperforms Frozen on both zero-shot and few-shot learning while using only 10% of multimodal data and a frozen T5-base LM backbone for pretraining.

2 Related Work

Recent work has shown impressive generalization and transfer ability of vision-language models that build on huge pretrained auto-regressive LMs (Alayrac et al., 2022; Tsimpoukelli et al., 2021; Eichenberg et al., 2022; Baevski et al., 2022). Frozen (Tsimpoukelli et al., 2021) updated an NFResNet encoder to create visual prefixes for the frozen LM, transferring the few-shot learning ability of the LM to a multimodal setting. MAGMA (Eichenberg et al., 2022) improved on the results of Frozen by incorporating adapter-based pretraining and a 25M image-text dataset including downstream data into pretraining. Luo et al. (2022) used cross-modal attention for encoding visual and text inputs. The Flamingo model (Alayrac et al., 2022) reached SOTA performance on few-shot VL tasks with a frozen CLIP encoder (Goh et al., 2021), while training a perceiver resampler and cross-attention with multimodal data on a LM objective. To reduce demands on supervised vision-text data, Wang et al. (2021c) pretrained a VL model from scratch using weak-labeled vision and text data. Despite various of parameter-efficient methods are applied during pretraining the models (Li and Liang, 2021; Morrone et al., 2019; Wang et al., 2021b; Kamath et al., 2020), these models are often of over billions of parameters and require high computing machines for deployment.

VL-T5 (Cho et al., 2021) is a moderate-size VL model, where the T5 backbone is updated on multitask objectives, with the encoder jointly learning from Faster-RCNN (Fu et al., 2021) features and input texts. FewVLM (Jin et al., 2021) improved on VL-T5 with prompt-based learning and simplified LM pretraining objectives. Sung et al. (2022) proposed adapter-based fine-tuning on downstream tasks. We exploit the potential of these moderate-size VL models and propose a more parameter-efficient few-shot learner with adapter-based pretraining.

3 Problem Statement

Despite that larger models are significantly more powerful following the scaling laws (Kaplan et al., 2020), we aim to answer the following key questions: i) how can we maximize the efficiency in transferring knowledge of a moderate-size frozen LM to a multimodal setting? ii) how much can we achieve on few-shot learning if we limit the size of data and trainable parameters in VL pretraining?

4 Method

This section describes MAP in details. Our approach is to maximize the knowledge transfer of a moderate-size LM to VL learning through adapter-based pretraining and prompting.

4.1 Model Architecture

We adopt a transformer-based encoder-decoder architecture (Parmar et al., 2017) to jointly encode vision and language inputs and generate target texts. As shown in Figure 2, the model is

Task	Input prompt	Example	
VQA	[Q]	**input:** What is this bird called?	**output:** parrot
	[Q] <text_1>	**input:** What is this bird called? <text_1>	**output:** parrot
	question: [Q] answer:	**input:** question: What is this bird called? answer:	**output:** parrot
	question: [Q] answer: <text_1>	**input:** question: What is this bird called? answer: <text_1>	**output:** parrot
Visual Entailment	[Q]	**input:** A hot air balloon is making a landing.	**output:** entailment
	[Q] <text_1>	**input:** A hot air balloon is making a landing. <text_1>	**output:** entailment
	hypothesis: [Q] label:	**input:** hypothesis: A hot air balloon is making a landing. label:	**output:** entailment
	hypothesis: [Q] label: <text_1>	**input:** hypothesis: A hot air balloon is making a landing. label:<text_1>	**output:** entailment

Table 1: Hand-crafted prompts. For VQA tasks, we prompt with "$question$:" for the input questions with "$answer$:" before the model output. A specific token "<$text$_1>" is used to indicate the generated words we expect (Jin et al., 2021). Similarly, we designed prompts of "$hypothesis$:" and "$label$:" for VE tasks.

mainly composed of three parts: a visual encoder, a transformer-based encoder-decoder LM backbone, and a series of adapter layers.

Visual Encoder Following Tsimpoukelli et al. (2021), we use a NFResNet encoder (Brock et al., 2021) to convert input images into visual embeddings. The visual embedding vectors then serve as prefixes to be jointly taken with text embeddings by the pretrained language model.

Encoder-decoder LM We adopt a moderate-size pretrained encoder-decoder LM, T5-base (Raffel et al., 2019), as the backbone of the model. The encoder builds joint representation of the input image-text pairs by taking the concatenated visual and text embeddings. Then, the decoder generates target texts in an auto-regressive manner.

Adapters Following Eichenberg et al. (2022), we use the bottleneck adapter modules, which are essential scaled residual bottleneck MLPs (Equation 1). The parameters of the adapters are updated instead of the entire model during pretraining. We add the adapters to the feed-forward and the attention blocks of the transformer following practical analysis by Eichenberg et al. (2022). In the transformer decoder, we experiment with different settings of adding the adapter layers to cross-attention or self-attention blocks.

$$A(h) = h + \lambda W^{up}\phi(W^{down}h) \quad (1)$$

4.2 Pretraining

Following Jin et al. (2021), we pretrain MAP on MaskedLM (Chang et al., 2018) and PrefixLM with paired image-caption data (Liu et al., 2019). However, instead of updating the entire parameter set of the LM, we only update the parameters of the adapter layers. We experiment with both settings of updating or freezing the visual encoder. Our adapter-based end-to-end pretraining is illustrated in Figure 2.

4.3 Few-shot Learning

In downstream tasks, we experiment with few-shot learning with both prompting and in-context learning. For prompting, we use hand-crafted prompts (Jin et al., 2021) and train the model with few-shot examples to minimize the negative-log-likelihood (Table 1). For in-context learning, we concatenate a series of image-text pairs in order as a multimodal prompt and expect the model to predict the target text given a visual query.

5 Experiments

5.1 Datasets

For pretraining, we combine image-caption pairs from MS COCO caption (Zitnick et al., 2015) and Visual Genome (VG) (Bernstein et al., 2017).* To explore the influence of different pretrained data size, we designed 3 versions of the pretraining data, with the corresponding number of VG region-caption pairs extracted from each image set to 2, 10 and 36. This leads to 0.3M, 0.8M and 4.2M image-caption pairs. We do not include any downstream dataset in pretraining.

We evaluate MAP's transfer ability on five downstream tasks, including VQAv2, OKVQA, GQA, and VizWiz for visual question answering, and SNLI-VE for language-image understanding.

5.2 Training Details

Training Settings For pretraining, we set batch size as 240 and pretrain with 30 epochs. We use learning rate 1e-4 with 5% linear warmup. We build the model with PyTorch and run on 8 A100 GPUs for around 5 days. For few-shot learning, we use learning rate 5e-5 with 5% linear warmup and

*As the annotated captions in Visual Genome are region descriptions, MAP directly takes the image drawn with the corresponding bounding boxes.

†We report the accuracy with MAGMA model using the NFResNet encoder.

‡592K(COCO)+36*108K(VG)

Method	$\|Data\|$	VQAv2	OK-VQA	GQA	SNLI-VE	VizWiz	$\|LM\|$
VL-T5$_{novqa}$ (Cho et al., 2021)	4.9M	31.8	12.7	19.6	-	-	224M
VL-T5$_{novqa}$ (Cho et al., 2021)	0.3M	0.1	0.0	0.0	-	-	224M
FEWVLM$_{base}$ (Jin et al., 2021)	4.9M	**48.2**	15.0	**32.2**	-	-	224M
FEWVLM$_{base}$ (Jin et al., 2021)	0.3M	16.8	9.9	13.2	-	-	224M
MAP$_{base}$	4.2M	40.4	**17.1**	27.2	33.1	25.6	224M*
MAP$_{small}$	0.8M	40.5	16.8	22.9	32.5	25.2	224M*
MAP$_{tiny}$	0.3M	38.0	15.7	22.1	**41.9**	**27.9**	224M*

Table 2: Few-shot (16-shot) evaluation results on VQAv2, OK-VQA, GQA, VizWiz and SNLI-VE. Compared to baseline models, MAP can be trained with much fewer pre-training data and parameters with minor downstream performance degradation. The * symbol indicates the parameters are frozen during pretraining.

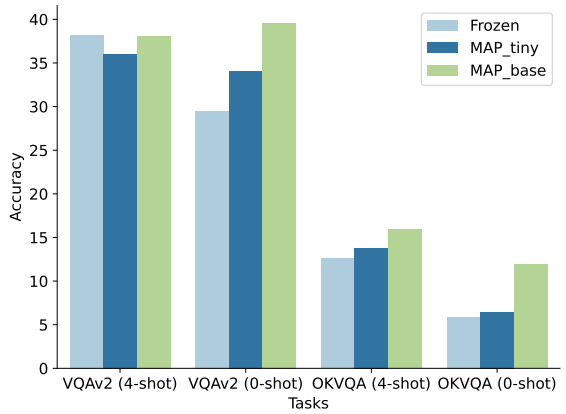

Figure 3: MAP outperforms Frozen on zero-shot VQAv2, zero-shot OK-VQA, and four-shot OK-VQA with a 30× lighter LM backbone (a 224M T5-base compared to a 7B GPT-style LM). Gains are retained even when using only 10% the size of pretraining data (0.3M for MAP$_{tiny}$ compared to 3.0M for Frozen.)

train for 200 epochs with the size of 16 for D_{train} and D_{dev}. We choose the best checkpoint for test set evaluation.

Hand-crafted Prompts As shown in Table 1, we use task-specific prompts designed for downstream evaluations to make the most of the transfer ability from the pre-trained model. We experiment of three different templates with corresponding input and result hints for VQA and VE tasks. Our prompts for VQA follows the design by Jin et al. (2021).

6 Evaluation and Results

To answer the questions that we raised in Section 3, we evaluate MAP on the aforementioned five downstream tasks in zero-shot and few-shot settings.

From our preliminary experiments, jointly updating the NFResNet vision encoder and adapter layers performs slightly better in pretraining than updating adapters only. We therefore applied such settings in all models pretrained using COCO combined with three versions of VG region-caption pairs, denoting as MAP$_{tiny}$, MAP$_{small}$ and MAP$_{base}$.

Transfer Efficiency To evaluate MAP's efficiency on transferring knowledge from a **frozen** LM to a multimodal setting, we compare MAP against Frozen on VQAv2 and OK-VQA. As shown in Figure 3, overall, MAP achieves better zero-shot and few-shot performance on both tasks. MAP$_{tiny}$ is able to outperform Frozen on zero-shot VQAv2, zero-shot and four-shot OK-VQA even with only 10% the size of pretraining data (0.3M v.s. 3.0M) and a 30× lighter LM backbone (224M v.s. 7B).

Data and Parameter Efficiency We compare MAP to fully trained VL models to evaluate how much can be achieved with limited pretraining data and number of trainable parameters. In Table 2, we show that compared to VL-T5 (Cho et al., 2021), on all the five downstream tasks, MAP$_{tiny}$ achieves much better results with only 16% in size of the pre-training dataset[†] and 48% in the number of trainable parameters. Moreover, MAP is strong at retaining its transfer learning ability while VL-T5 and FewVLM adapt the language modeling ability to the 0.3M pretraining data.

7 Conclusion and Future Work

We present an end-to-end moderate-size VL model, which surpasses Frozen and comparable-size fully trained baselines on few-shot learning over multiple image understanding tasks, while requiring much less training data and fewer parameters during pretraining. We expect to investigate the core transfer ability of pretrained VL models from a perspective beyond scaling. We propose open questions of whether data can be overloaded in pretraining and how can we use pretraining data more efficiently and wisely.

[†]Here we consider only the data on the LM objective. VL-T5 uses additional 3.3M multi-task data in pretraining.

Limitations

We experiment with two concatenation methods to build sequential VL inputs for in-context learning. However, we do not see improvements in few-shot performance with either settings, which may root in its pretraining strategy of taking only single image-caption pairs. Details are illustrated in Appendix A.

While our pretrained model obtains strong few-shot learning ability through parameter-efficient pretraining with a much smaller dataset, it is also possible that the small number of trainable parameters could limit its ability to learn from large-scale dataset. It is still an open question of how to automatically select multimodal data samples and maximizing data efficiency during the learning process.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. 2022. Flamingo: a visual language model for few-shot learning.

Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, Tadas Baltrusaitis, Amir Zadeh, Yao Lim, Ankur Bapna, Yu-An Chung, Nan Wu, Anmol Gulati, Andrew Brock, Soham De, Samuel Smith, Tom Brown, Benjamin Mann, Nick Ryder, Jared Subbiah, Prafulla Kaplan, Arvind Dhariwal, Pranav Neelakantan, Girish Shyam, Amanda Sastry, Sandhini Askell, and Ariel Agarwal. 2022. Prompting as multimodal fusing.

Michael S. Bernstein, Yuke Zhu, Oliver Groth, Joshua Kravitz, Ranjay Krishna, Kenji Hata, Li-Jia Li, David A. Shamma, Yannis Kalantidis, Justin Johnson, Li Fei-Fei, and Stephanie Chen. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*.

Andrew Brock, Soham De, Samuel L. Smith, and Karen Simonyan. 2021. High-performance large-scale image recognition without normalization. *international conference on machine learning*.

Ming-Wei Chang, Kenton Lee, Kristina Toutanova, and Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *north american chapter of the association for computational linguistics*.

Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying vision-and-language tasks via text generation. *arXiv: Computation and Language*.

Constantin Eichenberg, Sidney Black, Samuel Weinbach, Letitia Parcalabescu, and Anette Frank. 2022. Magma – multimodal augmentation of generative models through adapter-based finetuning.

Jianlong Fu, Dongmei Fu, Bei Liu, Zhicheng Huang, Zhaoyang Zeng, and Yupan Huang. 2021. Seeing out of the box: End-to-end pre-training for vision-language representation learning. *computer vision and pattern recognition*.

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

Woojeong Jin, Yu Cheng, Yelong Shen, Weizhu Chen, and Xiang Ren. 2021. A good prompt is worth millions of parameters? low-resource prompt-based learning for vision-language models. *arXiv: Computer Vision and Pattern Recognition*.

Aishwarya Kamath, Sebastian Ruder, Clifton Poth, Andreas Rücklé, Jonas Pfeiffer, Iryna Gurevych, Ivan Vulić, and Kyunghyun Cho. 2020. Adapterhub: A framework for adapting transformers. *empirical methods in natural language processing*.

Jared Kaplan, Samuel McCandlish, Thomas Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv: Learning*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *meeting of the association for computational linguistics*.

Peter J. Liu, Michael Matena, Katherine Lee, Adam Roberts, Yanqi Zhou, Noam Shazeer, Colin Raffel, Sharan Narang, and Wei Li. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*.

Ziyang Luo, Yadong Xi, Rongsheng Zhang, and Jing Ma. 2022. I-tuning: Tuning language models with image for caption generation.

Bruna Halila Morrone, Neil Houlsby, Sylvain Gelly, Mona Attariyan, Stanisław Jastrzębski, Andrei Giurgiu, Andrea Gesmundo, and Quentin de Laroussilhe. 2019. Parameter-efficient transfer learning for nlp. *international conference on machine learning*.

Niki Parmar, Lukasz Kaiser, Jakob Uszkoreit, Illia Polosukhin, Aidan N. Gomez, Ashish Vaswani, Noam Shazeer, and Llion Jones. 2017. Attention is all you need. *neural information processing systems*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*.

Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. 2022. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks.

Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *arXiv: Computer Vision and Pattern Recognition*.

Lijuan Wang, Zhe Gan, Xiaowei Hu, Yumao Lu, Zhengyuan Yang, Jianfeng Wang, and Zicheng Liu. 2021a. An empirical study of gpt-3 for few-shot knowledge-based vqa. *arXiv: Computer Vision and Pattern Recognition*.

Shean Wang, Yelong Shen, Zeyuan Allen-Zhu, Yuanzhi Li, Phillip Wallis, Weizhu Chen, and Edward J. Hu. 2021b. Lora: Low-rank adaptation of large language models. *arXiv: Computation and Language*.

Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. 2021c. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv: Computer Vision and Pattern Recognition*.

C. Lawrence Zitnick, Tsung-Yi Lin, Xinlei Chen, Saurabh Gupta, Ramakrishna Vedantam, Piotr Dollár, and Hao Fang. 2015. Microsoft coco captions: Data collection and evaluation server. *arXiv: Computer Vision and Pattern Recognition*.

A In-context Learning

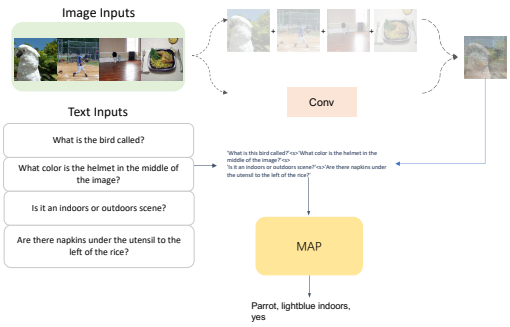

Figure 4: Concatenation Illustration

Our two approaches in concatenating inputs are illustrated in Figure 4. One is to mix-up images obtained by multiplying averaged weights within one glance and adding them all together with normalization, and the other way is to concatenate them on the channel dimension and pass through a convolution layer before feeding into the visual encoder.

B Modality Fusion

To validate that our joint-encoder's ability in learning multimodal representations, we apply linear probing on the representation output by the encoder and reach 66.4% in accuracy on the SNLI-VE task.