A Good Prompt Is Worth Millions of Parameters? Low-resource Prompt-based Learning for Vision-Language Models

Woojeong Jin[1*] **Yu Cheng**[2] **Yelong Shen**[2] **Weizhu Chen**[2] **Xiang Ren**[1]
[1]University of Southern California [2]Microsoft Corporation
{woojeong.jin,xiangren}@usc.edu {yu.cheng,yelong.shen,wzchen}@microsoft.com

Abstract

Large pretrained vision-language (VL) models can learn a new task with a handful of examples or generalize to a new task without fine-tuning. However, these gigantic VL models are hard to deploy for real-world applications due to their impractically huge model sizes and slow inference speed. In this work, we propose FEWVLM, a few-shot prompt-based learner on vision-language tasks. We pretrain a sequence-to-sequence Transformer model with both prefix language modeling (PrefixLM) and masked language modeling (MaskedLM), and introduce simple prompts to improve zero-shot and few-shot performance on VQA and image captioning. Experimental results on five VQA and captioning datasets show that FEWVLM outperforms Frozen (Tsimpoukelli et al., 2021) which is 31× larger than ours by 18.2% point on zero-shot VQAv2 and achieves comparable results to a 246× larger model, PICa (Yang et al., 2021). We observe that (1) prompts significantly affect zero-shot performance but marginally affect few-shot performance, (2) MaskedLM helps few-shot VQA tasks while PrefixLM boosts captioning performance, and (3) performance significantly increases when training set size is small.

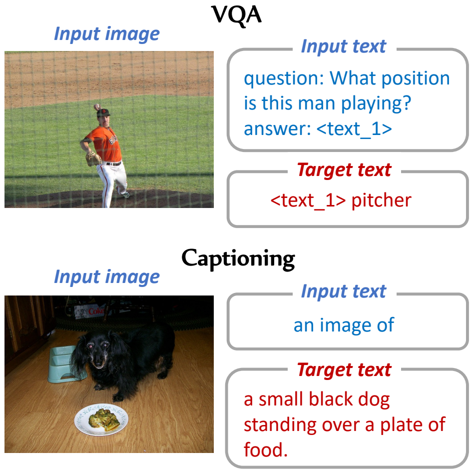

Figure 1: **Examples of few-shot tasks.** We use simple prompts in downstream tasks to improve zero-shot and few-shot performance.

1 Introduction

Fine-tuning large pretrained language models (PLMs) have led to strong results in various domains including vision-language tasks (Devlin et al., 2018; Raffel et al., 2019; Brown et al., 2020; Radford et al., 2021). Such large PLMs can learn a new task with a few examples or generalize to a new task without fine-tuning on any training examples, i.e., few-shot and zero-shot learning (Brown et al., 2020; Radford et al., 2021; Tsimpoukelli et al., 2021). Few-shot learning overcomes the challenges of data-hungry supervised learning, where collecting human-labeled data is costly and slow. However, recent few-shot models such as GPT3 (Brown et al., 2020), Frozen (Tsimpoukelli et al., 2021), and PICa (Yang et al., 2021) are too large to deploy in small or moderate computing machines due to their gigantic model sizes

In this paper, we study low-resource learning of VL tasks with moderate-size (comparatively smaller) vision-language models (VLMs), in which we fine-tune the model with no or a handful of training examples. This setup is more practical in that we don't need large computational resources to train and run the huge models and it is expensive to obtain a large number of quality training examples in the real world. In such a few-shot setting, task-specific prompts or task descriptions are important and have shown effectiveness in few-shot NLP tasks (Gao et al., 2020; Radford et al., 2021; Schick and Schütze, 2020a,b; Brown et al., 2020).

To extend the success to VL tasks, we study the following questions for prompt-based low-resource VL learning. Q1) How does prompt design affect zero-/few-shot learning on new tasks? Q2)

[*] Work was mainly done while interning at Microsoft Azure AI.

How do different pretraining objectives affect zero-/few-shot learning? Q3) How does the model's performance change with respect to the amount of training data? To answer these questions, we explore simple prompt formats on zero-/few-shot VL learning datasets. In addition, we study pretraining objectives on few-shot performance inspired by Raffel et al. (2019): prefix language modeling (PrefixLM) inspired by Raffel et al. (2019) and masked language modeling (MaskedLM). Lastly, we train our model with different training sizes in a few-shot manner and investigate the performance changes. To this end, we consider an encoder-decoder architecture (Vaswani et al., 2017) for few-shot VL tasks including visual question answering (Goyal et al., 2017; Marino et al., 2019; Hudson and Manning, 2019) and captioning (Agrawal et al., 2019; Young et al., 2014).

In our empirical analysis, our prompt-based few-shot learning shows comparable results to a huge vision-language model which is 246× larger than ours. We observe that (1) prompts significantly affect zero-shot performance but marginally affect few-shot performance, (2) MaskedLM helps few-shot VQA tasks while PrefixLM boosts captioning performance, and (3) performance sharply increases until a certain amount of training data in few-shot learning. We expect better zero-/few-shot capability with simple prompts to guide VLMs to generate the correct answer in diverse tasks.

2 Related Work

Vision-language few-shot learning. Recently, several few-shot learners on vision-language tasks were proposed including decoder-style architectures (Radford et al., 2019; Brown et al., 2020), Frozen (Tsimpoukelli et al., 2021) and PICa (Yang et al., 2021), and an encoder-decoder architecture (Dosovitskiy et al., 2020), SimVLM (Wang et al., 2021). Frozen (Tsimpoukelli et al., 2021) is a large language model based on GPT-2 (Radford et al., 2019), and is transformed into a multimodal few-shot learner by extending the soft prompting to incorporate a set of images and text. Their approach shows the few-shot capability on visual question answering and image classification tasks. Similarly, PICa (Yang et al., 2021) uses GPT-3 (Brown et al., 2020) to solve VQA tasks in a few-shot manner by providing a few in-context VQA examples. It converts images into textual descriptions so that GPT-3 can understand the images. SimVLM (Wang et al., 2021) adopts an encoder-decoder architecture and is trained with a prefix language modeling objective on weakly-supervised datasets. It demonstrates its effectiveness on a zero-shot captioning task While these models achieve improvement on few-shot tasks, they are impractical to use in real-world applications due to their gigantic model sizes.

Language model prompting. Providing prompts or task descriptions play an vital role in improving pretrained language models in many tasks (Gao et al., 2020; Radford et al., 2021; Schick and Schütze, 2020a,b; Brown et al., 2020). Among them, GPT models (Radford et al., 2019; Brown et al., 2020) achieved great success in prompting or task demonstrations in NLP tasks. However GPT models are too large to easily handle it on many applications. In light of this direction, prompt-based approaches improve small pretrained models in few-shot text classification tasks (Gao et al., 2020; Schick and Schütze, 2020a,b). CLIP (Radford et al., 2021) also explores prompt templates for image classification which affect zero-shot performance. We follow these core ideas so we aim to improve zero-shot and few-shot performance using prompts in vision-language tasks.

Pretraining vision-language models. There is a tremendous amount of work in training generic models for a variety of V+L task, such as visual question answering (VQA), and image captioning *etc*. Most of the existing methods employ BERT-like architectures (Devlin et al., 2018) to learn cross-modal representations from a concatenated sequence of visual region features and language token embeddings. For example, early efforts such as VisualBERT, VL-BERT and Oscar (Su et al., 2019; Li et al., 2019, 2020b) propose either a single-stream or two-stream Transformer-based framework. Chen et al. (2019) conduct comprehensive studies on the effects of different pre-training objectives on the learned representations. Cao et al. (2020) design a set of meticulously designed probing tasks to decipher the inner workings of multimodal pre-training. Gan et al. (2020) propose an adversarial learning framework to improve the vision-and-language representation. Instead of using the regional features extracted by pre-trained object detection models like Faster-RCNN, SOHO (Huang et al., 2021) proposes to jointly learn Convolutional Neural Network (CNN) and Transformer for cross-modal alignments from image-text pairs.

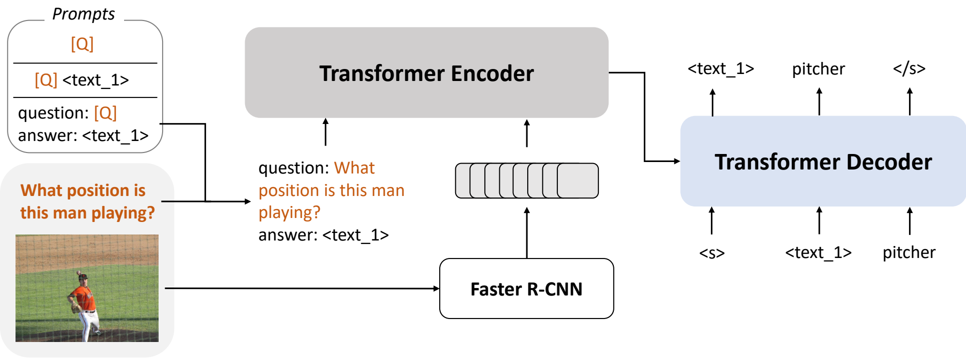

Figure 2: **Illustration of FewVLM.** This shows inference of FewVLM. We choose one of the prompts and feed the question with it and the given image; then the model will generate the answer.

The UNIMO architecture (Li et al., 2020a) can leverage the large scale of non-paired text corpus and image collections for cross-modal learning.

3 Analysis Setup

In this work, we study the zero-shot and few-shot performance of vision-language models $\mathcal{L}$. We introduce our analysis setup: problem formulation, analysis questions, downstream tasks and datasets, evaluation metrics, and baselines.

3.1 Problem Formulation

For zero-shot tasks, a pretrained VL model $\mathcal{L}$ have no access to training set $\mathcal{D}_{train}$ and development set $\mathcal{D}_{dev}$, and directly makes inference on the test instances $\mathcal{D}_{test}$. For few-shot tasks, we compose a dev set $\mathcal{D}_{dev}$ from training data and ensure that $|\mathcal{D}_{train}| = |\mathcal{D}_{dev}|$ following Perez et al. (2021); Gao et al. (2020) to tune the hyper-parameters and select the model. We limit the sizes of training and development sets to meet the goal of learning from limited data. The size of $\mathcal{D}_{train}$ and $\mathcal{D}_{dev}$ are small — i.e., we set the size of both to 16 in our study. We also test different sizes of training and validation set as an ablation study.

3.2 Analysis Questions

We aim to answer the following questions in this study through experiments on multiple VL datasets.

Q1) How does prompt design affect zero-/few-shot learning on new tasks? Providing a pre-trained language model with task-specific prompts or significantly improves zero-shot and few-shot performance on NLP domains (Gao et al., 2020; Schick and Schütze, 2020a,b; Brown et al., 2020). For this question, we test several simple prompts on vision-language tasks and analyze how large zero-shot and few-shot performance is affected by the different prompts in Sec. 5.3.

Q2) How do different pretraining objectives affect zero-/few-shot learning? We study two different pretraining objectives on few-shot performance: prefix language modeling (PrefixLM) inspired by Raffel et al. (2019) and masked language modeling (MaskedLM). In this setup, we pretrain our model with different objectives and test the model on zero-shot and few-shot tasks in Sec. 5.4.

Q3) How does the model's performance change with respect to the amount of training data? For this question, we investigate the effect of different training sizes for few-shot learning. We train the model with the different sizes of few-shot examples $\{0, 10, 20, 30, 40, 50, 60, 70, 80\}$ in Sec. 5.5.

3.3 Downstream Tasks and Datasets

In this work, we study two different tasks: a visual question answering and a captioning task. The visual question answering task requires models to answer a question to a given context image. We convert the visual question answering task into a generation task so that the model can produce the answers. The captioning task requires a model to generate descriptions for a given context image.

We include VQA v2.0 (Goyal et al., 2017), OK-VQA (Marino et al., 2019), and GQA (Hudson and Manning, 2019) for visual question answering tasks, and NoCaps (Agrawal et al., 2019), and Flickr30k (Young et al., 2014) for image captioning. We use Karpathy split (Karpathy and Fei-Fei, 2015) for Flickr30k, which re-splits train and val images into 29,000 / 1,014 / 1,000 for train / validation / test.

3.4 Evaluation Metrics

To evaluate few-shot performance, we randomly sample 5 different training and dev splits and measure average performance on the 5 splits. We train the vision-language models with 200 epochs for the few-shot setup and choose the best checkpoint on the dev set. We train the models with training data from COCO captioning for NoCaps following Wang et al. (2021) since NoCaps does not have its own training data. We evaluate on the VQAv2 validation set, GQA test-dev, OK-VQA test set, Karpathy split for COCO captioning, and NoCaps validation set. We adopt accuracy for VQA datasets and CIDEr (Vedantam et al., 2015) and SPICE (Anderson et al., 2016) as evaluation metrics for captioning.

3.5 Baselines

We include strong zero-/few-shot vision-language learners as our competitors: Frozen (Tsimpoukelli et al., 2021), PICa (Yang et al., 2021) for VQA datasets and SimVLM (Wang et al., 2021) for captioning datasets. Also, we compare them with fully fine-tuned models $\mathcal{L}_{full}$ as upper bounds of few-shot models for each task; these models are fine-tuned on the entire datasets while few-shot models can access a small amount of data. For fully fine-tuned models $\mathcal{L}_{full}$, we borrow numbers from Uniter$_{large}$ (Chen et al., 2019) for VQAv2, Oscar (Li et al., 2020b) for GQA, SimVLM (Wang et al., 2021) and VinVL (Zhang et al., 2021) for NoCaps CIDER and SPICE respectively, and Unified VLP (Zhou et al., 2020) for Flickr30k captioning. We include VL-T5$_{\text{no-vqa}}$ as a baseline which is pretrained without visual question answering datasets (Cho et al., 2021).

4 Prompt-based Low-resource Learning

In this section, we introduce preliminaries of the vision-languages models, pretraining objectives, and pretraining datasets. Then we present several prompt designs on downstream tasks for low-resource learning. Fig. 2 shows an illustration of FEWVLM.

4.1 Seq2Text Vision-language Models

We adopt an encoder-decoder architecture (Vaswani et al., 2017), VL-T5 (Cho et al., 2021), to encode visual and text inputs and generate target text. We represent an input image with 36 object regions from a Faster R-CNN (Ren et al., 2015) trained on Visual Genome (Krishna et al., 2017). These sets of region representations are fed into the encoder by appending them to the text. We train the model parameters θ by minimizing the negative log-likelihood of target text y tokens given input text x and image v:

$$L_\theta = -\sum_{i=1}^{|y|} \log P_\theta(y_i | y_{<i}, x, v) \qquad (1)$$

The models are free from extra model designs for new tasks such as introducing task-specific heads, so they are good options for few-shot tasks. Following Cho et al. (2021), we convert tasks into a text-to-text format; we formulate the task labels to corresponding text (Figs. 1 and 2), and we learn these different tasks by predicting label text with the language modeling objective (1). For downstream tasks, we don't introduce task-specific heads; a model generates the label text and we evaluate the model with the generated text by strict string matching.

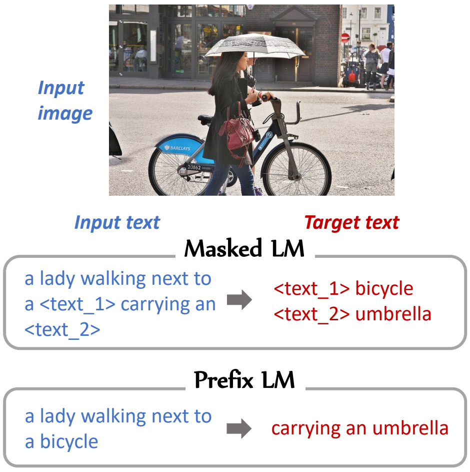

Figure 3: **Pretraining objectives.** We pretrain our model with masked language modeling (MaskedLM) and prefix language modeling (PrefixLM).

4.2 Pretraining Objectives

We pretrain the models with prefix language modeling (PrefixLM) and masked language modeling (MaskedLM). Fig. 3 illustrates the PrefixLM and MaskedLM.

Prefix language modeling. We include prefix language modeling (PrefixLM) following Raffel et al. (2019). Given an image and a span of text, this objective randomly splits the text into two separate components; the former component with the given image is used as inputs to the encoder and the latter

Table 1: **Prompts used in our experiments.** We test different input and target prompts on zero-shot and few-shot tasks. `[Q]` and `[A]` refer to question text and answer text, respectively. `[caption]` denotes caption text. `<text_1>` is a sentinel token. We append image features to input text.

Task	**Input prompt**	**Target prompt**	**Example**
VQA	`[Q]`	`[A]` or `<text_1> [A]`	**input:** What position is this man playing? **output:** `<text_1>` pitcher
	`[Q] <text_1>`	`[A]` or `<text_1> [A]`	textbfinput: What position is this man playing? `<text_1>` **output:** pitcher
	question: `[Q]` *answer:*	`[A]` or `<text_1> [A]`	**input:** question: What position is this man playing? answer: **output:** pitcher
	question: `[Q]` *answer:* `<text_1>`	`[A]` or `<text_1> [A]`	**input:** question: What position is this man playing? answer: `<text_1>` **output:** `<text_1>` pitcher
Captioning	no prompt	`[caption]`	**input:** *no input text* **output:** a small black dog standing over a plate of food.
	a picture of	`[caption]`	**input:** a picture of **output:** a small black dog standing over a plate of food.
	a photo of	`[caption]`	**input:** a photo of **output:** a small black dog standing over a plate of food.
	an image of	`[caption]`	**input:** an image of **output:** a small black dog standing over a plate of food.

component is used as target text to be generated by the decoder.

Masked language modeling. We follow Cho et al. (2021) to do masked language modeling. This objective is to replace random spans with numbered sentinel tokens, e.g., `<text_1>`, and then the masked text is fed into the encoder. Then the decoder generates the masked spans as target text. We randomly mask 15% of input text tokens and replace them with sentinel tokens.

4.3 Pretraining Data

We collect captioning data from MS COCO (Lin et al., 2014; Chen et al., 2015) and Visual Genome (VG) (Krishna et al., 2017) to pretrain models. Unlike VL-T5 (Cho et al., 2021), we do not include visual question answering datasets in pretraining, so we ensure that the model never learn the visual question answering tasks in pretraining. We also pretrain models with Conceptual Captions (Sharma et al., 2018) as we will see in our ablation study (Sec. 5.6).

4.4 Prompt-based Learning

In downstream tasks, we train our model with few-shot examples. Given a prompt template $\mathcal{P}$, we first get input text and target text using the template $x, y = \mathcal{P}(\text{input}, \text{label})$. Then we train model parameters by minimizing the negative log-likelihood in Eq. (1). In inference, we use the same prompt and let our model generate the label text. Here we obtain the final label by removing the target prompt template. Figs. 1 and 2 show an example of input and target text and illustration of FewVLM.

4.5 Prompt Design

Prompts affect the performance of the vision-language model (Cho et al., 2021); we study the effect of different prompts on the zero-shot and few-shot performance on downstream tasks. Table 1 shows prompts we used in our experiments.

Visual Question Answering. The visual question answering tasks (VQA, OK-VQA, and GQA) require models to answer a question to a given context image. Recent approaches (Chen et al., 2019; Tan and Bansal, 2019; Su et al., 2019; Li et al., 2019, 2020b) tackle visual question answering tasks as multi-label classification over a predefined set of answer candidates. Instead, we approach the visual question answering tasks as a generation task so that the model can produce the answers without introducing any task-specific heads. In this setup, prompts act as constraints to guide the models to generate proper formats of answers; models might generate a sentence for VQA, which is not the correct format, without prompts.

Therefore, we study several prompts for input and output as shown in Table 1. For input prompts, we explore four different templates: "`[Q]`" (a question), "*question:* `[Q]` *answer:*" and with the `<text_1>` sentinel token at the end. Similarly to masked language modeling, we expect models to generate words thanks to the sentinel token. For target prompts, we explore two different templates: "`[A]`" (an answer) and "`<text_1> [A]`" (an answer with a sentinel token). Here we also aim to mimic MaskedLM's target text format, so the similar format helps the model quickly adapt to the new task.

Captioning. In NoCaps and Flickr30k, we explore four different input prompts: no prompt, "*a picture of*", "*a photo of*", and "*an image of*". We study the effect of different word choices in this captioning task. While the three different words have similar meanings, they show different performance in zero-shot and few-shot tasks. For target prompts, we just train the model with the original caption without any additional prompts.

Table 2: **Zero-shot VQA results.** VL-T5$_{\text{no-vqa}}$ is pre-trained without VQA datasets. Compared to larger models, Frozen and PICa-Full, our models outperform them or show the comparable results.

Model	Model size	VQAv2	OK-VQA	GQA
VL-T5$_{\text{no-vqa}}$	224M	13.5	5.8	6.3
Frozen	7B	29.5	5.9	-
PICa	175B	-	**17.5**	-
FEWVLM$_{base}$	224M	43.4	11.6	27.0
FEWVLM$_{large}$	740M	**47.7**	*16.5*	**29.3**
Fine-tuned $\mathcal{L}_{full}$	-	72.6	-	61.5

Table 3: **Few-shot VQA results.** We report average performance over 5 different splits. The size of training and validation sets are 16 for our FEWVLM and VL-T5$_{\text{no-vqa}}$, and Frozen and PICa use 4 and 16 in-context training examples, respectively. For the fair comparison to Frozen, we include FEWVLM$^{*}_{base}$ with 4 training and validation examples.

Model	Model size	VQAv2	OK-VQA	GQA
VL-T5$_{\text{no-vqa}}$	224M	31.8	12.7	19.6
Frozen	7B	38.2	12.6	-
PICa	175B	**54.3**	**43.3**	-
FEWVLM$^{*}_{base}$	224M	45.1	14.5	26.9
FEWVLM$_{base}$	224M	48.2	15.0	32.2
FEWVLM$_{large}$	740M	*51.1*	*23.1*	**35.7**
Fine-tuned $\mathcal{L}_{full}$	-	72.6	-	61.5

Table 4: **Zero-shot captioning results.** We use the CIDEr and SPICE metrics for evaluation.

Model	Model size	NoCaps		Flickr30k	
		CIDEr	SPICE	CIDEr	SPICE
VL-T5$_{\text{no-vqa}}$	224M	4.4	5.3	2.6	2.0
SimVLM$_{huge}$	-	**101.4**	-	-	-
FEWVLM$_{base}$	224M	42.2	8.5	31.0	10.0
FEWVLM$_{large}$	740M	*47.7*	**9.1**	**36.5**	**10.7**
Fine-tuned $\mathcal{L}_{full}$	-	112.2	13.1	67.4	17.0

Table 5: **Few-shot captioning results.** We report average performance over 5 different splits. We use the CIDEr and SPICE metrics for evaluation.

Model	Model size	NoCaps		Flickr30k	
		CIDEr	SPICE	CIDEr	SPICE
VL-T5$_{\text{no-vqa}}$	224M	22.0	6.87	12.8	8.3
FEWVLM$_{base}$	224M	48.6	10.0	32.6	12.8
FEWVLM$_{large}$	740M	**53.1**	**10.4**	**37.0**	**13.5**
Fine-tuned $\mathcal{L}_{full}$	-	112.2	13.1	67.4	17.0

5 Results and Discussion

In this section, we first discuss our main results on zero-shot and few-shot tasks and then answer the questions we raised.

5.1 Experiment Details

For pretraining, we set batch size 1,280 and 800 for FEWVLM$_{base}$ and FEWVLM$_{large}$, respectively and pretrain them with 30 epochs. We use learning rate 1e-4 with 5% linear warmup. For few-shot learning, we train models with 200 epochs, learning rate 5e-5 and 5% linear warmup and choose the best checkpoint on the dev set. We use "question: [Q] answer <text_1>" and "<text_1>" for visual question answering prompts, and "an image of" and "[caption]" for captioning prompts for our method, which show the best performance. We will study the effect of different prompts in Sec 5.3. The sizes of of $\mathcal{D}_{train}$ and $\mathcal{D}_{dev}$ are 16.

5.2 Main Results

Zero-shot performance. We evaluate the existing models in a zero-shot manner, in which models do not have access to any training data. Tables 2 and 4 show the results on visual question answering datasets and caption datasets, respectively. First, FEWVLM with our prompts achieves better performance than other baselines on VQA datasets. In particular, our FEWVLM$_{base}$ significantly outperforms Frozen which is about 31× larger than ours. Also, PICa based on GPT3 (Brown et al., 2020) shows the best performance on OK-VQA. It is noticeable that our FEWVLM$_{large}$, the 246× smaller model, achieves the comparable result to PICa. Compared to VL-T5$_{\text{no-vqa}}$ which is the same architecture as ours, FEWVLM$_{base}$ improves VQAv2 performance by about 30% point. As we will see in the later section, our pretraining objectives and the prompts boost the VQA performance. On NoCaps, SimVLM$_{huge}$ shows the best performance. Our FEWVLM$_{base}$ significantly improves the performance compared to VL-T5$_{\text{no-vqa}}$ although they share the same architecture. The PrefixLM objective boosts the captioning performance which will be discussed in Sec. 5.4.

Few-shot performance. Tables 3 and 5 show the few-shot performance on VQA and captioning datasets. Note that the size of training and validation sets are 16 for our FEWVLM and VL-T5$_{\text{no-vqa}}$, and Frozen and PICa use 4 and 16 in-context training examples, respectively. We study different sizes

Table 6: **Prompt study on VQAv2.** We test different input and target prompts for few-shot learning. Note that the zero-shot setup does not require target prompts since we don't train models for zero-shot predictions.

Target \ Input	[Q]	[Q] <text_1>	*question:* [Q] *answer:*	*question:* [Q] *answer:* <text_1>
Zero-shot	3.7	9.9	19.0	43.4
[A]	47.0	47.4	44.7	45.2
<text_1> [A]	47.0	48.6	47.2	48.2

Table 7: **Prompt study on Flickr30k.** We test different input prompts for captioning with a CIDEr metric.

	no prompt	*a picture of*	*a photo of*	*an image of*
Zero-shot	9.6	15.2	25.6	31.0
Few-shot	32.0	31.1	31.8	32.6

Table 8: **Results on different pretraining objectives.** We test our pretraining objectives and how it affects zero-shot and few-shot performance. We train FewVLM$_{base}$ with 16 training and validation examples.

Objective	**VQAv2**	**GQA**	**Flickr30k CIDEr**
Zero-shot			
MaskedLM	42.4	25.1	4.6
PrefixLM	11.9	6.7	26.8
MaskedLM + PrefixLM	**43.4**	**27.0**	**31.0**
Few-shot			
MaskedLM	46.0	31.4	18.5
PrefixLM	40.8	27.6	31.8
MaskedLM + PrefixLM	**48.2**	**32.2**	**32.6**

of training and validation sets in Sec. 5.5.

On VQAv2 and OK-VQA, PICa shows the best performance while our FewVLM$_{large}$ achieves the comparable result on VQAv2. Note that our FewVLM$_{large}$, the 246× smaller than PICa. OK-VQA requires external knowledge to answer unlike other VQA datasets, so larger models and large pretraining data (prior knowledge) are necessary to improve. Interestingly, FewVLM$^{*}_{base}$, which is also trained with 4 training examples, outperforms Frozen which is 31× larger than ours. On captioning data, FewVLM$_{base}$ notably outperforms VL-T5$_{\text{no-vqa}}$ by 31.1% point on NoCaps CIDEr.

5.3 How does prompt design affect zero-/few-shot learning on new tasks?

In this section, we examine the effect of different prompts on FewVLM$_{base}$ in Tables 6 and 7. We test the model on VQAv2 and Flickr30k datasets. On VQAv2, different input prompts significantly affect zero-shot performance. For input prompts, <text_1> helps the zero-shot predictions significantly. We conjecture that <text_1> guides the model to predict masked spans similarly to MaskedLM, so it improves the performance. Interestingly, different input prompts do not affect the few-shot performance with a target prompt <text_1> [A]. Also the prompt <text_1> [A] provides more consistent accuracy than the prompt [A]. Similar to MaskedLM, the <text_1> in target prompts play an important role in guiding the model to predict spans.

On Flickr30k, we examine different word choices of prompts: "a picture of," "a photo of," "an image of." On the zero-shot setup, performance is affected remarkably by different input prompts. For instance, using "an image of" outperforms using no prompt by 57.2% point. It is noticeable that different word choices also affect the captioning results. On the few-shot setup, however, different prompts do not influence the captioning significantly. FewVLM without any prompt shows the comparable result to other prompts. The prompts are important when there is no training data, but models may quickly learn how to generate labels (captions in this case) for the certain task from a few examples.

5.4 How do different pretraining objectives affect zero-/few-shot learning?

We investigate how pretraining objectives affect different tasks. We pretrain FewVLM with different pretraining objectives: masked language modeling (MaskedLM) and prefix language modeling (PrefixLM).

As shown in Table 8, we observe that MaskedLM helps VQA tasks while PrefixLM helps captioning tasks in zero-shot and few-shot settings. We conjecture that MaskedLM is to predict spans, which is analogous to predict correct answers to questions, and PrefixLM is to generate the rest of the given prefix, which is similar to captioning tasks. In other words, if the pretraining task is similar to the downstream tasks, then it will help performance further. When pretraining with both objectives, they create a synergetic effect and thus increase the results more.

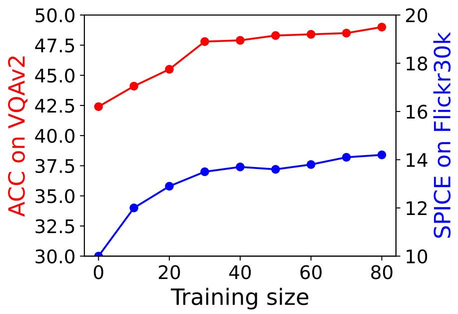

Figure 4: **Results on different training sizes.** We investigate different training sizes on few-shot tasks.

Table 9: **Pretraninig datasets.** We examine different pretraining datasets on each downstream tasks.

Dataset	**VQAv2**	**GQA**	**Flickr30k**
MS COCO, VG	48.2	32.2	32.6
Conceptual Captions	36.7	25.9	22.3

5.5 How does the model's performance change with respect to the amount of training data?

We show the results of different sizes of training data in Fig. 4. We observe that the performance sharply increases until a certain training size, e.g. 30 on VQAv2, and it slowly increases after that. It is interesting that in the few-shot setting, where models can access a limited number of data, comparatively larger training data does not significantly improve the performance further. We conjecture that FewVLM can learn the proper format to generate as label text with the comparatively smaller training size.

5.6 Pretraining Datasets

Here we pretrain our model with different datasets: MS COCO and Visual Genome (VG), and Conceptual Captions (CC). We investigate which pretraining dataset helps the downstream tasks in a few-shot manner. In our experiment, we observe that MS COCO and VG datasets are more helpful to the downstream tasks than CC.

6 Conclusion

In this work, we presented FewVLM, a few-shot prompt-based learner on vision-language tasks. On six different datasets, FewVLM outperforms baselines and shows comparable results to PICa which is 246× larger than ours. We observed that prompts are vital in zero-shot and few-shot tasks and each pretraining objective helps different few-shot tasks. Also, we found out that comparatively larger training data does not significantly improve performance in few-shot learning. Future work includes exploring automatic prompt generation and diverse formats of few-shot tasks such as multiple-choice VQA. Finding optimal prompts require exhaustive engineering to achieve the best performance. If we automatically find optimal prompts then we expect impressive results on diverse tasks. Also, this work focused on generative VQA and captioning tasks and hasn't explored other types of tasks such as multiple-choice VQA. We leave the exploration of these directions to future investigations.

Acknowledgement

We would like to thank Linjie Li and Yen-Chun Chen who have provided us some of the pretraining datasets.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8948–8957.

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*, pages 382–398. Springer.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Jize Cao, Zhe Gan, Yu Cheng, Licheng Yu, Yen-Chun Chen, and Jingjing Liu. 2020. Behind the scene: Revealing the secrets of pre-trained vision-and-language models. In *ECCV*.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Uniter: Learning universal image-text representations.

Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. 2021. Unifying vision-and-language tasks via text generation. *arXiv preprint arXiv:2102.02779*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. 2020. Large-scale adversarial training for vision-and-language representation learning. In *NeurIPS*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

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

Zhicheng Huang, Zhaoyang Zeng, Yupan Huang, Bei Liu, Dongmei Fu, and Jianlong Fu. 2021. Seeing out of the box: End-to-end pre-training for vision-language representation learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

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

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*.

Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. 2020a. Unimo: Towards unified-modal understanding and generation via cross-modal contrastive learning. *arXiv preprint arXiv:2012.15409*.

Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. 2020b. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer.

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. Springer.

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

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *arXiv preprint arXiv:2105.11447*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

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. *arXiv preprint arXiv:1910.10683*.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99.

Timo Schick and Hinrich Schütze. 2020a. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

Timo Schick and Hinrich Schütze. 2020b. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*.

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

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*.

Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *arXiv preprint arXiv:2106.13884*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

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

Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. 2021. Simvlm: Simple visual language model pretraining with weak supervision. *arXiv preprint arXiv:2108.10904*.

Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. 2021. An empirical study of gpt-3 for few-shot knowledge-based vqa. *arXiv preprint arXiv:2109.05014*.

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.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588.

Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13041–13049.

Table 10: **Model architectures.**

Hyperparameter	**FewVLM$_{base}$**	**FewVLM$_{large}$**
# Layers	12+12	24+24
Hidden dimension	768	1,024
FF hidden size	3,072	4,096
# Attention head	12	16
Attention head size	64	64

Table 11: **Prompt study on VQAv2.** We test different input and target prompts for few-shot learning. We use FewVLM with MaskedLM. Note that the zero-shot setup does not require target prompts since we don't train models for zero-shot predictions.

Target \ Input	`[Q]`	`[Q]` `<text_1>`	*question:* `[Q]` *answer:*	*question:* `[Q]` *answer:* `<text_1>`
Zero-shot	13.5	17.8	13.8	42.4
`[A]`	31.8	25.3	38.8	37.9
`<text_1> [A]`	43.2	45.6	44.7	46.0

Table 12: **Prompt study on Flickr30k captioning.** We test different input prompts for captioning. We use FewVLM with MaskedLM.

	no prompt	*a picture of*	*a photo of*	*an image of*
Zero-shot	4.6	5.3	4.6	4.6
Few-shot	18.4	18.7	16.9	18.5

A Model Architectures

Table 10 shows model parameters in our model, FewVLM. FewVLM$_{base}$ and FewVLM$_{large}$ is based on VL-T5 (Cho et al., 2021) and T5 (Raffel et al., 2019), respectively.

B Prompt Study on FewVLM with MaskedLM

Tables 11 and 12 show prompt study on FewVLM with MaskedLM. On VQAv2, different input prompts significantly affect zero-shot performance. Also the sentinel token `<text_1>` in target prompts boost the performance overall. However, the prompts on captioning do not affect the zero-shot performance a lot. This is because FewVLM with MaskedLM has a little ability on captioning tasks.