MAPL🍁: Parameter-Efficient Adaptation of Unimodal Pre-Trained Models for Vision-Language Few-Shot Prompting

Oscar Mañas[△] **Pau Rodriguez**[*,◇] **Saba Ahmadi**[*,△]
Aida Nematzadeh[♡] **Yash Goyal**[♣] **Aishwarya Agrawal**[△,†]
[△]Mila, Université de Montréal [◇]ServiceNow Research [♡]DeepMind
[♣]Samsung - SAIT AI Lab, Montreal [†]Canada CIFAR AI Chair
`oscar.manas@mila.quebec`

Abstract

Large pre-trained models have proved to be remarkable zero- and (prompt-based) few-shot learners in unimodal vision and language tasks. We propose MAPL, a simple and parameter-efficient method that reuses frozen pre-trained unimodal models and leverages their strong generalization capabilities in multimodal vision-language (VL) settings. MAPL learns a lightweight mapping between the representation spaces of unimodal models using aligned image-text data, and can generalize to unseen VL tasks from just a few in-context examples. The small number of trainable parameters makes MAPL effective at low-data and in-domain learning. Moreover, MAPL's modularity enables easy extension to other pre-trained models. Extensive experiments on several visual question answering and image captioning benchmarks show that MAPL achieves superior or competitive performance compared to similar methods while training orders of magnitude fewer parameters. MAPL can be trained in just a few hours using modest computational resources and public datasets. We release our code and pre-trained model weights at `https://github.com/oscmansan/mapl`.

1 Introduction

Over the past few years, natural language processing and computer vision have witnessed impressive progress in learning models capable of transferring to unseen tasks or benchmarks (Brown et al., 2020; Zhang et al., 2022; Radford et al., 2021; Jia et al., 2021). Recently referred to as foundation models (Bommasani et al., 2021), these can be adapted to a wide range of *unimodal* vision and language tasks without any additional training.

In this work, we study reusing such powerful *unimodal* foundation models for *multimodal* vision-language (VL) downstream tasks. In particular, we propose to connect a vision encoder, such as CLIP (Radford et al., 2021), to an autoregressive language model (LM), such as GPT (Radford et al., 2018, 2019; Brown et al., 2020), with minimal additional training on multimodal data. Our goal is to obtain a single VL model that can leverage the in-context learning abilities (Brown et al., 2020) of the pre-trained LM to generalize to unseen VL tasks from just a few examples.

One challenge in connecting vision encoders with LMs is aligning the visual and textual representation spaces. Recent works have approached this by adapting the LM to visual representations, either by fine-tuning the entire LM (Dai et al., 2022) or training adapter layers (Eichenberg et al., 2021; Alayrac et al., 2022). These systems are computationally expensive to train as they have a large number of learnable parameters (hundreds of millions to a few billions) and use large-scale multimodal training data. On the other hand, Frozen (Tsimpoukelli et al., 2021) keeps the LM frozen, thus learning $\sim 10\times$ less parameters than the above methods. However, it requires training a visual encoder from scratch, which is also computationally expensive.

Differently, we aim to reuse large pre-trained unimodal models while keeping them completely frozen and free of adapter layers. We present **MAPL** (**M**ultimodal **A**daptation of **P**re-trained vision and **L**anguage models), a simple and parameter-efficient VL model capable of tackling unseen VL tasks. MAPL learns a lightweight mapping between the representation spaces of pre-trained unimodal models. MAPL has orders of magnitude fewer parameters than previous methods (including Frozen) and can be trained in just a few hours. Moreover, MAPL's modularity makes it general-purpose and easily extensible to newer and/or better pre-trained models. We evaluate MAPL on various image captioning and visual question answering (VQA) benchmarks and compare with Frozen (Tsimpoukelli et al., 2021) in a controlled setup. MAPL significantly outperforms

[*] denotes equal contribution.

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 2523–2548
May 2-6, 2023 ©2023 Association for Computational Linguistics

Frozen and achieves competitive performance compared to other methods (Eichenberg et al., 2021; Dai et al., 2022) trained on comparably sized data.

We further investigate the parameter efficiency of MAPL by training on only 1% of multimodal data (thousands of examples); we call this setting *low-data* learning. We also study *in-domain* learning: training on image-text pairs from the same domain as the downstream task domains. We train MAPL directly on 100% and 1% of in-domain data for each downstream task, without first pre-training on large-scale domain-agnostic data. Thus, we train specialized versions of MAPL for each downstream domain. Such low-data and in-domain learning are particularly useful when it is difficult to pre-train on large-scale domain-agnostic data. We found MAPL to be more effective than Frozen trained under the same settings.

To summarize, our contributions are: 1) we introduce MAPL, a parameter-efficient method capable of tackling unseen VL tasks, which can be trained using only modest computational resources and public datasets; 2) we conduct extensive experiments spanning various image captioning and VQA benchmarks, demonstrating MAPL achieves superior or competitive performance compared to similar methods while training orders of magnitude fewer parameters; and 3) we further investigate the parameter-efficiency of MAPL in two settings: low-data and in-domain. Our experiments show that MAPL is more effective than the considered methods in both settings.

2 Related Work

Fine-tuning based VL methods. A popular family of VL methods are based on the pre-training + fine-tuning paradigm. These methods are either encoder-only (Lu et al., 2019; Tan and Bansal, 2019; Chen et al., 2019; Li et al., 2020; Zhang et al., 2021) or encoder-decoder methods (Cho et al., 2021; Wang et al., 2021; Jin et al., 2022; Li et al., 2021) and use transformer-based architectures. These transformers are first pre-trained on domain-agnostic image-text pairs (e.g., Conceptual Captions (Sharma et al., 2018)) using self-supervised objectives, and then fine-tuned for each downstream task (e.g., VQA, image captioning). More recent models that are designed specifically for the task of image captioning use large pre-trained LMs (e.g., GPT-2 (Radford et al., 2019)) and fine-tune these models with image-caption pairs (Chen et al., 2021; Mokady et al., 2021; Luo et al., 2022). While all these approaches yield state-of-the-art performance for the tasks they are fine-tuned on, the learned model weights are highly specialized for a single task and cannot transfer to new tasks with zero or few examples. Differently, MAPL reuses the same set of weights for all downstream tasks without any additional training.

Few-shot learning based VL methods. Most similar to MAPL are methods that tackle unseen VL tasks in a zero/few-shot manner, by leveraging the in-context learning abilities of large pre-trained LMs (e.g., GPT-3 (Brown et al., 2020). These methods connect a vision encoder with a pre-trained LM to tackle VL tasks. Some methods (Dai et al., 2022; Hao et al., 2022) achieve this connection by fine-tuning the entire LM on image-text data, while others only train adapter layers inserted into the LM (Eichenberg et al., 2021). The vision encoder is pre-trained and kept frozen in both cases. Concurrent work Flamingo (Alayrac et al., 2022) pushes this idea even further by scaling up the amount of training data and the LM size. While inserting adapter layers requires training fewer parameters compared to fine-tuning the entire LM, the number of trainable parameters is still >100M; in contrast, MAPL only has 3.4M trainable parameters. Additionally, inserting adapter layers is not straightforward since it requires modifying the computational graph of the LM; MAPL only adds an external mapping network, which is easier to incorporate on top of pre-trained models. On the other hand, Frozen (Tsimpoukelli et al., 2021) keeps the pre-trained LM frozen and instead trains a vision encoder from scratch. This approach does not scale well with larger vision encoders (Sec. 4.5). MAPL keeps both the vision encoder and the LM frozen (thus further reducing the number of trainable parameters) and only learns a lightweight mapping network to connect both frozen models. Similar to MAPL, concurrent work LiMBeR (Merullo et al., 2022) also proposes to connect a frozen vision encoder with a frozen LM but using a linear mapping, which is not as parameter- and compute-efficient as MAPL (Sec. 4.5).

Mapping networks. MAPL trains a mapping network to align the visual and textual representations of the visual encoder and the LM, respectively. The architecture of our mapping network has some similarities with that in ClipCap (Mokady et al., 2021) and the Perceiver Resampler in Flamingo (Alayrac

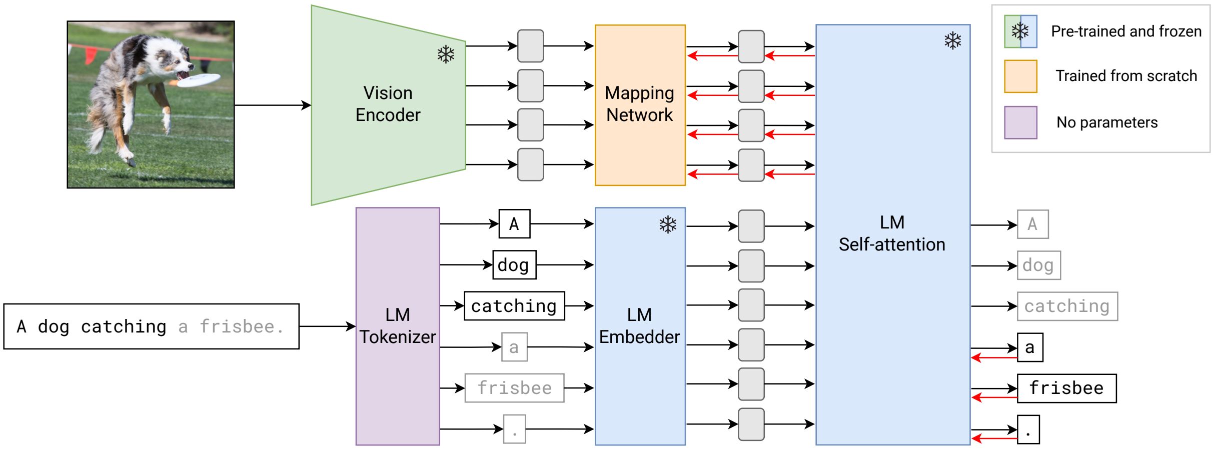


Figure 1: MAPL leverages a pre-trained vision encoder and a pre-trained LM, and learns a small *mapping network* to convert visual features into token embeddings. During training, only the mapping network is updated, keeping the vision encoder and the LM frozen (red arrows indicate gradient flow). At inference time, the system can take as input an arbitrary sequence of interleaved images and text, and generates free-form text as output.

et al., 2022). They all share a core transformer stack and a fixed number of learned constant embeddings. However, MAPL's mapping network is specifically designed to be parameter-efficient while maintaining expressivity (Sec. 3.1), containing only 3.4M parameters – orders of magnitude fewer than ClipCap's (43M) and Flamingo's (194M).

3 Method

MAPL is a vision-language (VL) multimodal model capable of generating text from a combination of visual and textual inputs. Our model builds on top of pre-trained vision-only and language-only models and leverages their strong generalization capabilities (e.g., zero-shot transfer, in-context learning) to tackle unseen VL tasks. MAPL is agnostic to the choice of these pre-trained unimodal models as long as they show such capabilities (Sec. 4.5). Concretely, MAPL maps the image representations from a vision encoder's output embedding space to a LM's token embedding space, so that the LM can be conditioned both on visual and textual information. To this end, we train a *mapping network* with an image captioning objective (Sec. 3.1, 3.2), while keeping the weights of the vision encoder and the LM frozen. Once the mapping network is trained, MAPL can be prompted with a few examples of unseen VL tasks and predict the response via text generation (Sec. 3.3). The overall model architecture is depicted in Figure 1.

3.1 Architecture

Pre-trained vision encoder. The vision encoder extracts a compact representation from an image. We use a CLIP (Radford et al., 2021) pre-trained vision encoder, which is trained on web-scale data and has shown strong zero-shot transfer capabilities to unseen image domains. In particular, we use CLIP's ViT-L/14 backbone (Dosovitskiy et al., 2020) since we empirically found it yields the best downstream VL performance among all variants. We use the flattened grid of spatial features (16×16) before the final projection layer and the representation corresponding to the [class] token, resulting in a sequence of $L_i = 257$ vectors of dimensionality $D_i = 1024$ each. This sequence of vectors is then fed to the mapping network.

Pre-trained autoregressive language model. Given an input text, the language model (LM) predicts its most likely completion by generating free-form text. For our LM, we use a pre-trained GPT-J model (Wang and Komatsuzaki, 2021) [1], a publicly-released 6B-parameter autoregressive LM trained on the Pile dataset (Gao et al., 2020). We chose this LM due to its strong in-context learning abilities, similar to that of GPT-3 (Brown et al., 2020) (which is not publicly available). The LM takes as input a text string, which is first divided into a sequence of discrete tokens by the LM's tokenizer. Each token is then individually transformed into a continuous embedding (of size $D_o = 4096$) by the LM's embedder. The sequence of token embeddings is fed to the self-attention layers in the LM's transformer block (using causal attention), which outputs a sequence of categorical distributions over the token vocabulary. Finally, a decoding mechanism generates free-form text from these distributions (greedy

[1] We also experiment with an OPT model, see Sec. 4.5.

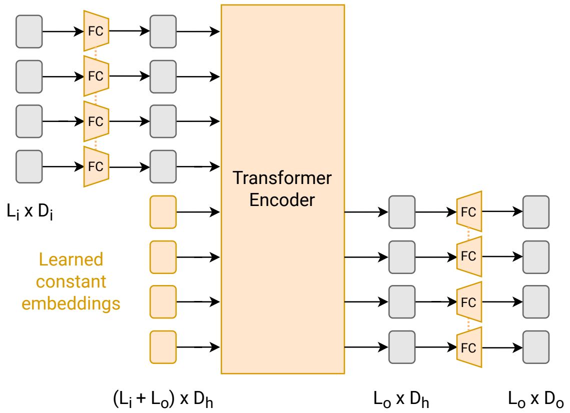


Figure 2: The mapping network takes a flattened grid of L_i visual features of dimension D_i each from the vision encoder and transforms it into a sequence of token embeddings of length L_o and dimension D_o, where D_o is the token embedding dimension of the LM. Note that the parameters are shared across fully-connected (FC) layers, on both sides of the encoder transformer.

decoding in our case).

Mapping network. The mapping network transforms a sequence of visual features from the vision encoder to a sequence of continuous embeddings which can be consumed by the LM's transformer. We design our mapping network considering the trade-off between expressivity (to learn a good mapping) and parameter count. Our architecture is based on a transformer encoder with 4 layers and 8 heads each. This transformer could directly take a sequence of projected visual features (from D_i to D_o) and output a sequence of embeddings of size D_o. However, in order to keep a low parameter count, we decouple the transformer hidden size D_h from the visual feature size D_i and the LM embedding size D_o by introducing a dimensionality bottleneck (Figure 2). In particular, each visual feature is first linearly projected from $D_i = 1024$ to $D_h = 256$ using a set of fully-connected (FC) layers. This sequence of projected features is then fed to the transformer, and the output representations are linearly projected from D_h to $D_o = 4096$ using another set of FC layers. To further reduce the parameter count of our mapping network, we share parameters across all FC layers in each set.

Yet another idea we use in our mapping network is to decouple the output sequence length of the transformer (L_o) from the input sequence length (L_i). We do this to obtain a much smaller $L_o = 32$ compared to $L_i = 257$, in order to reduce the computational complexity in the subsequent LM's self-attention layers, which in turn speeds up training and inference time. To achieve this decoupling, inspired by DETR (Carion et al., 2020), we concatenate a small and fixed number (L_o) of learned constant embeddings with the input sequence of the transformer and only use the output representations corresponding to these constant embeddings (Figure 2). Note that these output representations are conditioned on the input visual features via cross-attention in the transformer. The resulting mapping network architecture is shown in Figure 2. In total, our mapping network contains only 3.4M parameters. Since this is the only trainable component of our model, MAPL has orders of magnitude fewer total trainable parameters than existing methods such as Frozen (40.3M) or Flamingo (10.2B).

3.2 Training

Following previous works (Tsimpoukelli et al., 2021; Eichenberg et al., 2021), we train our model using a standard language modeling objective on image captions with teacher forcing (Lamb et al., 2016), i.e., we minimize the negative log-likelihood of the reference captions under the LM conditioned on the corresponding images. We only train the mapping network (from scratch) while keeping the vision encoder and the LM entirely frozen. This preserves the pre-trained models' capabilities while making the system modular and parameter-efficient. Even though the LM's weights are kept frozen, gradients are still back-propagated through its self-attention layers to train the mapping network.

3.3 Zero- and Few-shot Evaluation

Once the mapping network is trained, MAPL can tackle unseen VL tasks by prompting the LM with a combination of visual and textual inputs. We study zero-shot transfer to unseen image captioning benchmarks and few-shot transfer (via in-context learning) to the unseen task of visual question answering (VQA). For image captioning, we simply feed the mapped image embedding to the LM and start generating a caption. For zero-shot VQA, following Tsimpoukelli et al. (2021), we feed the mapped image embedding followed by the text `"Please answer the question. Question: {question} Answer:"`[2] and start generating the answer. For n-shot VQA, we select n support examples $(image, question, answer)$ from the training set at random, and prepend them to the

[2] Here `{question}` indicates a placeholder which gets replaced by the corresponding question in each example. Same applies to `{answer}` in the few-shot setting.

query; for each support example, we concatenate the mapped image embedding with the text "`Please answer the question. Question: {question} Answer: {answer}`".

4 Experiments

4.1 Experimental settings

Evaluation benchmarks. We evaluate MAPL on several VL benchmarks spanning VQA and image captioning. Note that our model is never trained for the task of VQA. For VQA, we evaluate on the validation splits of VQAv2 (Goyal et al., 2017), OK-VQA (Marino et al., 2019), TextVQA (Singh et al., 2019) and VizWiz-VQA (Gurari et al., 2018), and report performance using VQA accuracy (after the standard normalization (Antol et al., 2015)). For image captioning, we evaluate on the Karpathy-test split (Karpathy and Fei-Fei, 2015) of COCO Captions (Chen et al., 2015), and the validation splits of Conceptual Captions (CC) (Sharma et al., 2018)[3], TextCaps (Sidorov et al., 2020) and VizWiz-Captions (Gurari et al., 2018), and report performance using the BLEU@4, ROUGE-L, METEOR, CIDEr and SPICE metrics.

Training settings. We consider two settings to train our mapping network: domain-agnostic and in-domain training (described below). For each of these settings, we also study low-data learning by training our model on randomly sampled subsets of 1% training image-text pairs. Such low-data learning is useful when it is difficult to train models on large-scale data due to constraints on compute resources, data availability, etc.

For **domain-agnostic training**, we use the CC dataset, which is gathered by automatically scraping images and their corresponding alt-text fields from web pages. Thus, this dataset is not as clean as manually-curated datasets such as COCO Captions (e.g., the caption may not describe the image). Nevertheless, due to its large size (3.3M) and great diversity, it is the most commonly used dataset for domain-agnostic pre-training of VL models. However, for our model – having orders of magnitude less trainable parameters than other methods –, we observed the negative effect of noise in CC to be stronger than the positive effect of its large size (Sec. 4.4). Therefore, we train MAPL on a filtered version of CC (CC-clean) consisting of the top 398K most similar image-text pairs ranked by CLIP's image-text similarity score.[4] For completeness, we also report MAPL's performance when trained on the unfiltered CC dataset.

For **in-domain training**, we use image-caption pairs that come from the same domain as the downstream task domains, i.e., they have similar image and language distributions as those in the downstream datasets. For the image captioning downstream task, this amounts to the IID setting. The in-domain image captioning and VQA dataset pairs we consider are shown in Table 1. Each pair uses the same set of images, and focuses on the same set of VL skills; for instance, scene understanding (COCO Caps and VQAv2), reading and reasoning about text in images (TextCaps and TextVQA), understanding images captured by visually-impaired users (VizWiz-Caps and VizWiz-VQA), thus leading to similar image and language distributions across image-captioning and VQA. We train MAPL on both 100% and 1% of in-domain image-caption data and evaluate on all downstream benchmarks (including out-of-domain ones, e.g., VizWiz-VQA when trained on COCO Caps). Such in-domain training can be useful when it is difficult to first train on large-scale domain-agnostic data and then adapt to in-domain data by either fine-tuning or few-shot prompting.

	VQAv2	OK-VQA	TextVQA	VizWiz-VQA
COCO Caps	✓	✓		
TextCaps			✓	
VizWiz-Caps				✓

Table 1: In-domain dataset pairs.

Training details. For Conceptual Captions, TextCaps and VizWiz-Captions, we carve out a minival split consisting of 6% of training examples and train on the remaining 94%; for COCO Captions, we use the Karpathy-val split as minival. We use the AdamW optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and a weight decay of 0.01. The learning rate is increased linearly from 0 to 3×10^{-4} (7×10^{-4} for OPT-based models) over the first 1500 steps (15 for 1% of data) and kept constant for the rest of training. We use a batch size of 128 and we do early stopping based on the minival loss. We do not add any special tokens at the beginning of sentence, as GPT-J was not trained with `<BOS>` tokens. In order to fit a 6B-parameter LM into GPU memory, we use DeepSpeed ZeRO (Rajbhan-

[3] Due to broken image URLs, we only managed to download 13K out of 15K validation images.

[4] We selected a threshold on CLIP's similarity score such that the size of the filtered dataset is comparable to the size of manually curated datasets such as COCO Captions.

	Trainable params	Training examples	n-shot VQAv2			n-shot OK-VQA			n-shot TextVQA			n-shot VizWiz-VQA			n-shot Overall		
			0	4	8	0	4	8	0	4	8	0	4	8	0	4	8
Existing methods using domain-agnostic training																	
Frozen	40.3M†	3.3M	29.50	38.20	-	5.90	12.60	-	-	-	-	-	-	-	-	-	-
MAGMA $_{\text{CC12M}}$	243M†	3.8M	36.90	45.40	-	13.90	23.40	-	-	-	-	5.60	10.60	-	-	-	-
VLKD $_{\text{CC3M}}$	406M	3.3M	38.60	-	-	10.50	-	-	-	-	-	-	-	-	-	-	-
LiMBeR-CLIP‡	12.6M†	3.3M	33.33	40.34	-	-	-	-	-	-	-	-	-	-	-	-	-
Flamingo‡	10.2B	>2.1B	-	-	-	50.60	57.40	57.50	35.00	36.50	37.30	-	-	-	-	-	-
100% domain-agnostic training																	
MAPL-blind $_{\text{CC-clean}}$	3.4M	374K	20.62	35.01	35.11	4.84	14.68	14.28	3.68	5.43	5.82	3.18	8.65	9.55	8.08	15.94	16.19
Frozen* $_{\text{CC-clean}}$	40.3M	374K	25.98	37.80	38.52	5.51	18.86	19.91	5.11	6.15	6.30	4.33	11.28	16.68	10.23	18.52	20.35
MAPL $_{\text{CC-clean}}$	3.4M	374K	**33.54**	**45.13**	**45.21**	**13.84**	**24.25**	**23.93**	**8.26**	**8.88**	**8.77**	**11.72**	**18.46**	**19.52**	**16.84**	**24.18**	**24.36**
1% domain-agnostic training																	
Frozen* $_{\text{CC-clean}}$	40.3M	3.7K	26.22	36.69	37.41	5.50	**18.76**	**20.51**	5.71	**7.19**	7.53	3.83	**11.71**	**16.66**	10.31	**18.58**	**20.53**
MAPL $_{\text{CC-clean}}$	3.4M	3.7K	**30.80**	**37.38**	**37.95**	**8.77**	18.18	19.15	**6.40**	7.07	**7.74**	**5.68**	9.26	10.58	**12.91**	17.97	18.85
100% in-domain training																	
PICa*	0	0	20.61	46.86	47.80	11.84	**31.28**	**33.07**	-	-	-	-	-	-	-	-	-
Frozen* $_{\text{COCO}}$	40.3M	414K	32.09	38.90	39.42	9.81	20.72	21.83	7.54	6.82	6.74	5.87	12.07	17.35	13.82	19.63	21.33
Frozen* $_{\text{TextCaps}}$	40.3M	103K	32.49	37.39	38.03	11.34	19.87	20.82	8.83	7.33	7.51	6.25	12.26	16.86	14.73	19.21	20.80
Frozen* $_{\text{VizWiz}}$	40.3M	110K	26.93	37.38	37.91	5.85	19.12	20.64	6.38	7.44	7.47	5.57	13.06	18.06	11.18	19.25	21.02
MAPL $_{\text{COCO}}$	3.4M	414K	**43.51**	**48.75**	**48.44**	**18.27**	31.13	31.63	10.99	11.10	11.08	**14.05**	17.72	19.18	21.70	**27.17**	**27.58**
MAPL $_{\text{TextCaps}}$	3.4M	103K	38.83	43.34	43.43	16.33	25.07	25.92	**22.27**	**19.53**	**19.75**	12.31	16.69	18.18	**22.43**	26.15	26.82
MAPL $_{\text{VizWiz}}$	3.4M	110K	32.80	42.94	43.20	11.70	24.91	25.73	9.27	10.36	10.23	10.42	**20.63**	**23.10**	16.05	24.71	25.56
1% in-domain training																	
Frozen* $_{\text{COCO}}$	40.3M	4.1K	30.18	37.23	37.89	9.33	19.60	20.71	7.43	7.65	7.67	4.37	12.00	16.48	12.83	19.12	20.69
Frozen* $_{\text{TextCaps}}$	40.3M	1.0K	32.09	36.72	37.25	10.75	18.85	19.51	8.17	7.57	7.28	5.39	11.79	16.20	14.10	18.73	20.06
Frozen* $_{\text{VizWiz}}$	40.3M	1.1K	29.62	37.30	37.87	7.57	19.36	20.60	7.16	7.17	7.25	4.53	**12.51**	**17.56**	12.22	19.08	**20.82**
MAPL $_{\text{COCO}}$	3.4M	4.1K	**37.69**	**40.42**	**40.84**	**13.92**	**21.66**	**22.41**	8.30	6.96	6.84	**6.94**	10.72	12.43	**16.71**	**19.94**	20.63
MAPL $_{\text{TextCaps}}$	3.4M	1.0K	33.57	36.70	36.87	12.46	17.45	18.21	**9.34**	**8.29**	**8.62**	6.54	9.58	11.62	15.48	18.00	18.83
MAPL $_{\text{VizWiz}}$	3.4M	1.1K	31.88	36.81	37.04	9.59	17.64	17.64	7.25	5.99	6.04	4.73	9.48	11.33	13.36	17.48	18.01

Table 2: Evaluation on few-shot VQA. For MAGMA $_{\text{CC12M}}$ and VLKD $_{\text{CC3M}}$, we report their best results when training only on domain-agnostic data (CC12M and CC3M, respectively). (†) indicates our informed estimation. (‡) indicates concurrent work.

dari et al., 2020) stage 2 optimizations. Freezing the LM's weights also brings massive savings in GPU memory during training, as fine-tuning with an Adam-based optimizer would require at least 4× GPU memory to store gradients, average, and squared average of the gradients. The whole system was trained on 4 A100 (40GB) GPUs for about 4 hours (for the CC-clean dataset). Unless otherwise stated, we repeat the experiments with two different random seeds and report the average performance.

Existing methods and baselines. We report the performance of several baselines and existing methods. First, to verify that the LM in MAPL is not ignoring the visual input, inspired by Tsimpoukelli et al. (2021), we train a blind version of MAPL (MAPL-blind) where the input images are replaced with zeros but the mapping network weights are still trained (to serve as prompt-tuning for the LM). Second, to estimate the upper-bound on how well we can do in VQA by representing images with text (rather than with continuous embeddings), we evaluate PICa (Yang et al., 2021), which directly prompts the LM with image captions, followed by questions for VQA. We reimplement PICa (denoted PICa*) using MAPL's LM (and evaluate on VQAv2 and OK-VQA using ground-truth COCO captions) for controlled comparison. Third, we compare MAPL with Frozen (Tsimpoukelli et al., 2021), as this is the most similar method to ours that also uses a frozen LM. We reimplement Frozen (denoted Frozen*) using MAPL's LM for controlled comparison. Lastly, we report the performance of other methods similar to MAPL: MAGMA (Eichenberg et al., 2021), VLKD (Dai et al., 2022), LiMBeR (Merullo et al., 2022), ClipCap (Mokady et al., 2021) and the published numbers from Frozen (Tsimpoukelli et al., 2021).[5] Note that all these methods (unless otherwise noted) are trained on domain-agnostic data, so we only compare with MAPL trained on CC-clean. For completeness, we also report results from Flamingo (Alayrac et al., 2022), which has orders of magnitude more learnable parameters than MAPL and is trained on considerably more data.

4.2 Evaluation of domain-agnostic learning

We report few-shot VQA results in Table 2 and image captioning results in Table 3. Subscripts in the first column denote the training dataset. *Overall* accuracies denote average of per-benchmark accuracies. First, we see that MAPL $_{\text{CC-clean}}$ substantially outperforms MAPL-blind $_{\text{CC-clean}}$ both on VQA and image captioning, proving that the visual inputs are not ignored by the LM in MAPL. Second, we find that MAPL $_{\text{CC-clean}}$ outperforms Frozen* $_{\text{CC-clean}}$ by a considerable margin on all VL benchmarks (with overall accuracy improvements of +6.61% 0-shot and +5.66% 4-shot on VQA tasks,

[5] We only add results which are reported on the same dataset splits as in MAPL.

	Trainable params	Training examples	CC B@4	CC CIDEr	COCO B@4	COCO CIDEr	TextCaps B@4	TextCaps CIDEr	VizWiz-Caps B@4	VizWiz-Caps CIDEr	Overall B@4	Overall CIDEr
Existing methods using domain-agnostic training												
ClipCap $_{\text{CC3M}}$	43M	3.3M	-	71.82	-	-	-	-	-	-	-	-
VLKD $_{\text{CC3M}}$	406M	3.3M	-	-	18.20	61.10	-	-	-	-	-	-
100% domain-agnostic training												
MAPL-blind $_{\text{CC-clean}}$	3.4M	374K	0.35	5.05	2.75	5.75	1.35	2.15	1.50	1.80	1.49	3.69
Frozen* $_{\text{CC-clean}}$	40.3M	374K	2.45	22.60	5.25	13.90	2.65	4.60	2.05	2.65	3.10	10.94
MAPL $_{\text{CC-clean}}$	3.4M	374K	**6.75**	**79.75**	**12.30**	**54.30**	**5.80**	**22.95**	**4.95**	**20.95**	**7.45**	**44.49**
1% domain-agnostic training												
Frozen* $_{\text{CC-clean}}$	40.3M	3.7K	0.75	6.55	3.05	5.25	1.70	1.65	1.50	1.40	1.75	3.71
MAPL $_{\text{CC-clean}}$	3.4M	3.7K	**1.75**	**19.65**	**5.80**	**17.85**	**2.70**	**5.40**	**2.15**	**4.85**	**3.10**	**11.94**
100% in-domain training												
Frozen* $_{\text{COCO}}$	40.3M	414K	0.65	9.05	20.05	61.35	6.95	11.75	5.45	6.20	8.28	22.09
Frozen* $_{\text{TextCaps}}$	40.3M	103K	0.20	3.55	4.05	6.70	8.85	16.95	4.40	5.25	4.38	8.11
Frozen* $_{\text{VizWiz}}$	40.3M	110K	0.25	4.40	3.75	6.05	4.10	5.65	19.00	76.85	6.78	23.24
ClipCap $_{\text{COCO}}$	43M	414K	-	-	33.53	113.08	-	-	-	-	-	-
MAPL $_{\text{COCO}}$	3.4M	414K	**2.25**	**34.50**	**36.45**	**125.20**	16.60	41.40	18.00	41.35	**18.33**	**60.61**
MAPL $_{\text{TextCaps}}$	3.4M	103K	0.90	13.05	9.80	28.65	**18.35**	**62.55**	11.20	31.85	10.06	34.03
MAPL $_{\text{VizWiz}}$	3.4M	110K	0.90	18.80	13.55	48.35	11.35	31.20	**34.70**	**141.30**	15.13	59.91
1% in-domain training												
Frozen* $_{\text{COCO}}$	40.3M	4.1K	0.25	3.60	6.20	12.80	2.80	3.15	2.85	2.30	3.03	5.46
Frozen* $_{\text{TextCaps}}$	40.3M	1.0K	0.10	2.60	1.65	2.80	3.65	5.00	2.00	2.25	1.85	3.16
Frozen* $_{\text{VizWiz}}$	40.3M	1.1K	0.20	3.40	2.90	3.20	3.35	3.45	12.70	40.55	4.79	12.65
MAPL $_{\text{COCO}}$	3.4M	4.1K	**0.80**	**12.10**	**19.65**	**65.90**	7.00	12.85	6.20	9.60	**8.41**	**25.11**
MAPL $_{\text{TextCaps}}$	3.4M	1.0K	0.30	3.90	4.10	8.05	**8.35**	**16.90**	5.00	7.25	4.44	9.03
MAPL $_{\text{VizWiz}}$	3.4M	1.1K	0.20	3.90	2.95	4.80	3.45	5.05	**18.40**	**71.10**	6.25	21.21

Table 3: Evaluation on image captioning. For VLKD $_{\text{CC3M}}$, we report their best results when training only on domain-agnostic data (CC3M).

	Training examples	VQAv2 4-shot	OK-VQA 4-shot	TextVQA 4-shot	VizWiz-VQA 4-shot	CC CIDEr	COCO CIDEr	TextCaps CIDEr	VizWiz-Caps CIDEr	Overall 4-shot	Overall CIDEr
Frozen* $_{\text{CC-clean}}$	0.4M	37.79	**19.29**	**6.25**	**11.11**	22.70	14.00	5.00	2.70	**18.61**	11.10
Frozen* $_{\text{CC-cleanish}}$	1.0M	**37.82**	18.49	6.12	10.16	37.60	20.60	6.60	3.20	18.15	17.00
Frozen* $_{\text{CC}}$	2.7M	37.81	18.33	5.56	9.97	**57.60**	**22.20**	**8.00**	**4.20**	17.92	**23.00**
MAPL $_{\text{CC-clean}}$	0.4M	44.35	24.03	**9.65**	17.33	72.70	**54.60**	**23.80**	**21.10**	23.84	43.05
MAPL $_{\text{CC-cleanish}}$	1.0M	**46.63**	**25.99**	8.48	**19.65**	88.30	54.10	22.30	19.80	**25.19**	**46.13**
MAPL $_{\text{CC}}$	2.7M	43.26	20.96	5.20	19.31	**101.10**	44.10	16.70	15.90	22.18	44.45

Table 4: Impact of data quality and size. These experiments are run with one seed only.

+4.35 BLEU@4 and +33.55 CIDEr on image captioning tasks). Importantly, this is achieved while training an order of magnitude fewer parameters (3.4M vs 40.3M). Next, MAPL $_{\text{CC-clean}}$ is competitive compared to existing methods (MAGMA, VLKD, ClipCap) and concurrent work LiMBeR, despite training one-two orders of magnitude fewer parameters on significantly less multimodal data. Lastly, MAPL $_{\text{CC-clean}}$'s performance is still far from the performance of Flamingo, which trains orders of magnitude more parameters on orders of magnitude more data. However, we believe MAPL to be an effective method for scenarios with constrained computational resources. For MAPL's qualitative results, see App. 4.6.

Low-data learning. When trained on only 1% domain-agnostic data, MAPL $_{\text{CC-clean}}$ outperforms Frozen* $_{\text{CC-clean}}$ for all image captioning evaluations (by +1.35 BLEU@4 and +8.23 CIDEr, overall) and all 0-shot VQA evaluations (by +2.60% overall accuracy), while achieving competitive performance on 4- and 8-shot VQA evaluations. In summary, these results show the effectiveness of our method in low-data settings, highlighting its usefulness for applications where data is scarce.

4.3 Evaluation of in-domain learning

In Tables 2 and 3, we observe that both MAPL and Frozen* benefit from directly training on in-domain data, compared to few-shot transfer from large-scale domain-agnostic pretraining. For instance, MAPL $_{\text{COCO}}$ and Frozen* $_{\text{COCO}}$ respectively outperform MAPL $_{\text{CC-clean}}$ and Frozen* $_{\text{CC-clean}}$ on VQAv2, OK-VQA and COCO Captions when trained on 100% of data. Interestingly, this performance gap is larger for MAPL compared to Frozen* by +2% 0-shot accuracy and +3.77% 4-shot accuracy averaged across VQAv2 and OK-VQA, and +9.35 BLEU@4 and +23.45 CIDEr on COCO Captions. A similar trend can be observed for TextCaps and TextVQA. Surprisingly, for 0-shot VQA and image captioning, training

	Vision encoder	Language model	Mapping network	VQAv2		OK-VQA		TextVQA		VizWiz-VQA		CC	COCO	TextCaps	VizWiz-Caps	Overall		
				0-shot	4-shot	0-shot	4-shot	0-shot	4-shot	0-shot	4-shot	CIDEr	CIDEr	CIDEr	CIDEr	0-shot	4-shot	CIDEr
Frozen*	NF-ResNet-50	GPT-J	Transformer	27.98	36.66	5.88	18.44	4.56	7.87	3.67	10.32	21.79	14.87	5.42	3.03	10.52	18.32	11.28
Frozen*	ViT-L/16	GPT-J	Transformer	27.82	36.60	5.64	16.26	3.67	5.27	4.70	11.77	13.19	8.77	2.88	2.35	10.46	17.48	6.80
Frozen*	ViT-L/14	GPT-J	Transformer	24.45	36.03	4.14	16.27	3.91	5.33	3.27	10.09	13.89	8.27	2.66	2.50	8.94	16.93	6.83
MAPL	CLIP-ViT-L/14	GPT-J	Transformer	**33.54**	**45.13**	13.84	24.25	8.26	**8.88**	11.72	**18.46**	**79.75**	54.30	22.95	**20.95**	**16.84**	**24.18**	**44.49**
MAPL	IN-NF-ResNet-50	GPT-J	Transformer	28.12	40.86	10.86	21.66	6.15	7.01	6.40	14.22	39.64	32.99	12.41	9.55	12.88	20.94	23.65
MAPL	IN-ViT-L/16	GPT-J	Transformer	31.70	43.75	11.13	**25.50**	6.16	7.40	8.93	16.45	56.33	45.80	17.12	16.28	14.48	23.28	33.88
Frozen*	NF-ResNet-50	OPT-6.7B	Transformer	30.16	32.72	8.10	13.79	5.44	6.81	7.15	6.77	25.40	17.80	6.90	4.00	12.71	15.02	13.53
MAPL	CLIP-ViT-L/14	OPT-6.7B	Transformer	23.26	33.95	**15.27**	16.25	**8.90**	6.41	**15.40**	9.47	66.60	**54.40**	**23.30**	19.60	15.71	16.52	40.98
MAPL	CLIP-ViT-L/14	GPT-J	Linear	30.55	37.09	12.20	16.69	7.02	5.80	8.81	12.49	60.00	43.80	18.30	13.70	14.65	18.02	33.95
MAPL	CLIP-ViT-L/14	GPT-J	MLP	28.99	43.69	11.07	25.33	6.60	8.39	9.73	17.14	70.40	49.10	20.90	20.30	14.10	23.64	40.18

Table 5: Ablation studies. We assess the impact of the choice of vision encoder (top), LM (middle) and mapping network architecture (bottom). All models are trained on 100% of CC-clean with a single seed. IN stands for ImageNet pre-training.

on just 1% of in-domain data outperforms 100% CC training for all benchmarks (except VizWiz-VQA) and both models. These results demonstrate the benefits of in-domain learning. When comparing MAPL vs. Frozen*, we observe that MAPL outperforms Frozen* for all tasks and benchmarks (except VizWiz-VQA) under both 100% and 1% in-domain settings. In fact, MAPL trained on just 1% in-domain data outperforms Frozen* trained on 100% in-domain data by +3.41% 0-shot accuracy and +1.14% 4-shot accuracy averaged across VQAv2, OK-VQA and TextVQA. Thus, MAPL is more effective than Frozen* at in-domain learning.

Contrary to the above trends, we observe that MAPL $_{\text{VizWiz}}$ under 1% in-domain training performs worse than MAPL $_{\text{COCO}}$ or MAPL $_{\text{TextCaps}}$ when evaluated on VizWiz-VQA. We hypothesize the visual embeddings extracted from CLIP's vision encoder for VizWiz images are not as good as those for COCO or TextCaps' images because the distribution of images in VizWiz (captured by visually-impaired people) is rather different from the distribution of images CLIP is trained on (scraped from the web), whereas for COCO and TextCaps this isn't the case. When training MAPL's mapping network on only 1% of VizWiz data, we believe the data is not large enough to compensate for the OOD pretrained vision encoder, so MAPL trained on COCO/TextCaps performs better on VizWiz-VQA. For in-domain training with 100% of data and 4/8-shot VQA, the mapping network has enough data to learn from and compensate for the OOD phenomenon. On the other hand, Frozen* does not suffer from this issue because its vision encoder is trained from scratch, allowing it to adapt to the image distribution.

Lastly, we observe that MAPL $_{\text{COCO}}$ outperforms ClipCap $_{\text{COCO}}$ by +2.92 BLEU@4 and +12.12 CIDEr on COCO Captions. MAPL $_{\text{COCO}}$ also outperforms PICa* (which represents images with ground-truth COCO captions) on VQAv2 and 0-shot OK-VQA, and achieves competitive results on few-shot OK-VQA; this demonstrates representing images with continuous embeddings is beneficial over caption-based image representations. Overall, we see that in-domain learning is beneficial and MAPL is more effective at it than similar methods.

4.4 Impact of data quality and size

To measure the impact of noise in the training data, we additionally train MAPL and Frozen* on the *full* CC dataset, consisting of 2.8M[6] examples, as well as on a *clean-ish* version consisting of the 1.0M most similar image-text pairs. In Table 4, we observe Frozen* achieves similar performance on few-shot VQA tasks when trained on noisy vs. clean data; however, Frozen*'s performance on image captioning decreases when trained on cleaner but smaller data. In contrast, MAPL generally benefits from cleaner training data, with the exception of evaluation on CC. We hypothesize both models perform better on CC when trained on larger (yet noisier) data because the CC validation set is IID with the full (noisy) CC training set. In the case of Frozen*, as we move away from the IID setting, the benefits from more data start diminishing (CC captioning > other captioning tasks > VQA tasks). For MAPL, the benefit from reduced noise in training data exceeds the degradation caused by a smaller data size, thanks to the reduced number of trainable parameters. These trends align with previous observations that larger models are more robust to noisy training data since they have enough capacity to model both noise and the desired function (Rolnick et al., 2017), while smaller models are more sample-efficient (Vapnik and Chervonenkis, 2015), i.e. they need less (clean) data to train effectively. Note that although MAPL's overall performance is higher when training on 1.0M than on 0.4M examples, we decided to train with 0.4M examples because training on ~2.5× more data (1.0M instead of 0.4M) required ~5× more iterations (always

[6] This is not the full 3.3M CC due to broken URLs.

early-stopping based on validation loss). So we did not think the slight performance increase due to more data was worth the ∼5× longer training time, especially because we were operating under a limited compute budget.

4.5 Ablation studies

In this section, we evaluate how the choice of vision encoder, LM and mapping network architecture impact MAPL's performance, and compare it with corresponding versions of Frozen* (where applicable). Results are presented in Table 5. Please refer to App. A.5 for more ablations.

First, to assess the impact of the choice of vision encoder, we train additional versions of MAPL replacing the CLIP pre-trained vision encoder (ViT-L/14 – 303M parameters) with encoders pre-trained on ImageNet: NF-ResNet-50 (23.5M) and ViT-L/16 (303M), and compare their performance with corresponding versions of Frozen*. We observe that: 1) MAPL outperforms Frozen* for each configuration of vision encoder, suggesting that MAPL is **robust to the choice of vision encoder's pre-training data and architecture**; and 2) Frozen* 's performance drops with bigger vision encoders (likely due to more trainable parameters), whereas MAPL improves due to the use of stronger pre-trained encoders. Thus, training the vision encoder from scratch (Frozen*) has limited application, while MAPL's **performance scales alongside the pre-trained vision encoder**.

Next, to evaluate the impact of the choice of LM, we train both MAPL and Frozen* replacing GPT-J by OPT-6.7B (Zhang et al., 2022). We see that in all settings except 0-shot VQAv2, MAPL outperforms Frozen*. See App. A.2 for discussion on 0-shot VQAv2 results. This suggests that MAPL is **robust to the choice of LM**. The above results also highlight how MAPL's **modularity** allows to easily replace the pre-trained vision encoder or the LM.

Lastly, to assess the impact of the choice of mapping network architecture, we replace the proposed transformer-based mapping network with two simpler architectures – a linear layer and a 2-layer MLP (see App. A.4 for details). We observe both these versions generally underperform the original setting (transformer-based), highlighting the **effectiveness of the proposed design**. We also note that in these simpler versions, the parameter count is directly proportional to the vision encoder's representation size and LM's embedding size, whereas in MAPL we decouple this using a dimensionality bottleneck (Sec. 3.1), making our mapping network **more parameter-efficient by design**.

4.6 Qualitative results

Figure 3 shows some selected samples from the web illustrating our interface at inference time using MAPL $_{\text{CC-clean}}$. The first two columns show successful results while the last column shows failure cases. For image captioning (top row), success cases show MAPL can generate meaningful and detailed textual descriptions of the scene. For zero-shot VQA (bottom row), success cases indicate that MAPL is able to parse the question and connect visual information to encyclopedic knowledge contained in the pre-trained LM. However, MAPL's visio-linguistic understanding is evidently still far from being perfect. More qualitative results (both success and failure cases) are provided in App. A.6.

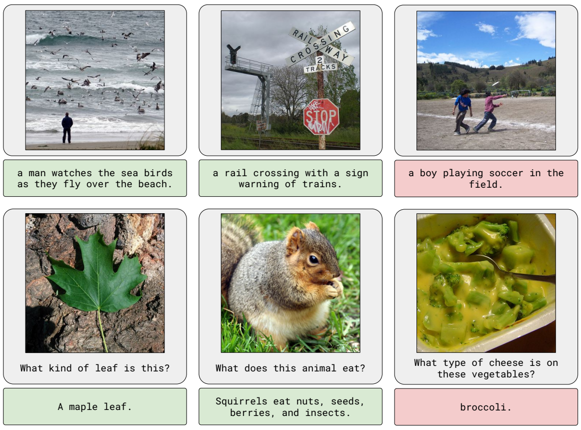

RAILWAY CROSSING
2
TRACKS
STOP
a man watches the sea birds as they fly over the beach.
a rail crossing with a sign warning of trains.
a boy playing soccer in the field.
What kind of leaf is this?
What does this animal eat?
What type of cheese is on these vegetables?
A maple leaf.
Squirrels eat nuts, seeds, berries, and insects.
broccoli.

Figure 3: Qualitative samples from the web using MAPL $_{\text{CC-clean}}$. (Multimodal) input is in gray, and MAPL's output is in green (success) or red (failure).

5 Conclusion

We introduce MAPL, a simple and parameter-efficient method to repurpose pre-trained and frozen unimodal models for multimodal tasks. Our experiments demonstrate that MAPL achieves superior or competitive performance compared to similar methods on several VL benchmarks while training orders of magnitude fewer parameters. Importantly, we also show that MAPL is effective in the low-data and in-domain settings thanks to its reduced number of trainable parameters. We leave as future work exploring training on a weighted mixture of image-text datasets, evaluating on more downstream tasks such as NLVR2 (Suhr et al., 2019) and Visual Dialog (Das et al., 2017), and investigating the use of masked LMs (Schick and Schütze, 2021; Chung et al., 2022) with MAPL.

Limitations

MAPL achieves reasonable performance on VL tasks, but it is still far from the performance of recent methods leveraging large-scale data and compute. On the other hand, MAPL is a preferable alternative in scenarios with constrained computational resources.

We observed our mapping network is sensitive to initialization, so different random seeds can yield non-negligible variance in downstream performance. We think this might be related to the reduced number of trainable parameters. We tried to reduce the effect of this variance by reporting average performance across different seeds. We also observed MAPL struggles to leverage more shots for in-context learning. We hypothesize this could be caused by our model being trained on single image-caption pairs – as opposed to the sequences of multiple images and texts seen during few-shot transfer, so a better pretext task might help (see App. A.1 for further discussion).

MAPL builds on top of pre-trained vision-only and language-only models, inheriting their capabilities but also their limitations. An important risk is that our model might inherit the existing social, gender or racial biases of pre-trained models. However, our limited qualitative analysis (see App. A.8) shows that providing visual information significantly changes the prior answer distribution of the LM. Therefore, how much of the underlying bias is retained remains an empirical question.

Ethics Statement

Model recycling. MAPL reuses vision-only and language-only foundation models. Hence, the expensive computational resources used to train these models can be amortized to help reduce energy and carbon costs.

Public datasets. MAPL is trained uniquely on publicly available datasets, which facilitates reproducibility and provides transparency on the origin and the characteristics of the data the model has seen.

Undesired biases. MAPL could be exposed to undesired biases from different sources. The pre-trained vision encoder might have been trained with data where certain races or genders are underrepresented, hence biasing our representation of images. The pre-trained LM might also be biased towards generating toxic or offensive language when fed with certain prompts. Finally, the image-text data used to align the representation spaces of such models was annotated by humans, so it might reflect a biased view of the world.

Broader impact. This work shows how one can easily adapt pre-trained vision encoders and LMs for multimodal tasks. Given the parameter-efficiency of our method, we believe it should be of great interest to the sections of the community that do not have access to large compute resources (e.g., small academic labs and independent researchers), and for low-data applications. While MAPL can be applied in many useful applications (e.g., aiding visually-impaired people), it also makes it simpler to create malicious or offensive multimodal systems from existing unimodal models. Further research efforts are needed on how to safely deploy such systems so that their behavior always aligns with ethical values.

Acknowledgements

We thank Pauline Luc, Siva Reddy, Lisa Anne Hendricks and Chris Dyer for their constructive feedback. We are grateful to the Mila IDT team for their technical support with the computational infrastructure. During this project, Aishwarya Agrawal was supported by the Canada CIFAR AI Chair award. We would also like to thank Samsung Electronics Co., Ldt. for funding this research.

References

Aishwarya Agrawal, Ivana Kajić, Emanuele Bugliarello, Elnaz Davoodi, Anita Gergely, Phil Blunsom, and Aida Nematzadeh. 2022. Rethinking evaluation practices in visual question answering: A case study on out-of-distribution generalization. *arXiv preprint arXiv:2205.12191*.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. 2021. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

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.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer.

Jun Chen, Han Guo, Kai Yi, Boyang Li, and Mohamed Elhoseiny. 2021. Visualgpt: Data-efficient adaptation of pretrained language models for image captioning. *arXiv preprint arXiv:2102.10407*.

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. In *International Conference on Machine Learning*, pages 1931–1942. PMLR.

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

Wenliang Dai, Lu Hou, Lifeng Shang, Xin Jiang, Qun Liu, and Pascale Fung. 2022. Enabling multimodal generation on CLIP via vision-language knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2383–2395, Dublin, Ireland. Association for Computational Linguistics.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017. Visual dialog. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 326–335.

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. In *ICLR*.

Constantin Eichenberg, Sidney Black, Samuel Weinbach, Letitia Parcalabescu, and Anette Frank. 2021. Magma–multimodal augmentation of generative models through adapter-based finetuning. *arXiv preprint arXiv:2112.05253*.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

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.

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

Yaru Hao, Haoyu Song, Li Dong, Shaohan Huang, Zewen Chi, Wenhui Wang, Shuming Ma, and Furu Wei. 2022. Language models are general-purpose interfaces. *arXiv preprint arXiv:2206.06336*.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pages 4904–4916. PMLR.

Woojeong Jin, Yu Cheng, Yelong Shen, Weizhu Chen, and Xiang Ren. 2022. A good prompt is worth millions of parameters: Low-resource prompt-based learning for vision-language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2763–2775, Dublin, Ireland. Association for Computational Linguistics.

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.

Alex M Lamb, Anirudh Goyal ALIAS PARTH GOYAL, Ying Zhang, Saizheng Zhang, Aaron C Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. *Advances in neural information processing systems*, 29.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in Neural Information Processing Systems*, 34.

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

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.

Ziyang Luo, Yadong Xi, Rongsheng Zhang, and Jing Ma. 2022. VC-GPT: visual conditioned GPT for end-to-end generative vision-and-language pre-training. *CoRR*, abs/2201.12723.

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.

Jack Merullo, Louis Castricato, Carsten Eickhoff, and Ellie Pavlick. 2022. Linearly mapping from image to text space. *arXiv preprint arXiv:2209.15162*.

Ron Mokady, Amir Hertz, and Amit H Bermano. 2021. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*.

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. In *International Conference on Machine Learning*, pages 8748–8763. PMLR.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

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.

Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory optimizations toward training trillion parameter models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE.

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–7.

David Rolnick, Andreas Veit, Serge Belongie, and Nir Shavit. 2017. Deep learning is robust to massive label noise. *arXiv preprint arXiv:1705.10694*.

Timo Schick and Hinrich Schütze. 2021. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352.

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.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. In *NAACL-HLT (2)*.

Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. 2020. Textcaps: a dataset for image captioning with reading comprehension. In *European Conference on Computer Vision*, pages 742–758. Springer.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8317–8326.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2019. A corpus for reasoning about natural language grounded in photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428.

Hao Tan and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111.

Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34.

Vladimir N Vapnik and A Ya Chervonenkis. 2015. On the uniform convergence of relative frequencies of events to their probabilities. In *Measures of complexity*, pages 11–30. Springer.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax.

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*.

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.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

A Appendix

A.1 Leveraging more shots

In Table 2, we observe MAPL's performance rapidly plateaus as the number of few-shot examples increases beyond 4. We hypothesize this could be related to the mapping network being trained on single image-caption pairs, and/or the visual embeddings still not being fully in-distribution with the language embeddings. Intuitively, a handful of examples may often help with task location (Reynolds and McDonell, 2021); however, the more shots are added, the more out-of-distribution the multimodal prompt becomes. This issue could be mitigated with in-context example selection (Yang et al., 2021) or better mixing of visual and textual modalities.

A.2 0-shot VQAv2 results with OPT

In Table 5, we observe Frozen*-OPT outperforms MAPL-OPT on 0-shot VQAv2. Upon close inspection, we notice MAPL-OPT often generates longer answers for yes/no questions, which receive a score of 0 according to VQA accuracy and VQAv2 reference answers – this is a problem of the metric and not the model itself (Agrawal et al., 2022). After filtering all answers starting with "yes" or "no" to leave only the short answer, MAPL-OPT achieves a VQA accuracy of 40.14% while Frozen*-OPT only reaches 32.03%.

A.3 4-shot results with OPT

In Table 5, we also observe that few-shot VQA performance is considerably lower for configurations using OPT-6.7B as language model. This is possibly due to the lack of a relative positional encoding (Shaw et al., 2018) in OPT, which is required for the transformer to generalize to prompt sequences where an image is not always in the first absolute position, or which contain more than one image (Tsimpoukelli et al., 2021).

A.4 Implementation details on simpler mapping networks

In Sec. 4.5, we ablate the choice of mapping network architecture and replace it by simpler architectures. Similarly to Eichenberg et al. (2021); Merullo et al. (2022), the linear mapping is applied per-position on top of a flattened grid of visual features, and it projects from $D_i = 1024$ to $D_o = 4096$ dimensions (4.2M parameters). The output sequence length L_o is thus equal to $L_i =$ 257 (instead of 32) – as explained in Sec. 3.1, this increases the computational complexity in the subsequent LM, which in turn increases training and inference time considerably. Similarly to Mokady et al. (2021), the 2-layer MLP is applied on top of a global vector of visual features. The MLP's hidden dimensionality D_h is equal to $D_i = 1024$, and the output dimensionality is $D_o = 32 * 4096$, which we split into 32 vectors of 4096 dimensions (135.3M parameters).

A.5 Additional ablation studies

Table 6 shows the results of our additional ablation studies. Unless specified otherwise, we perform all ablations on MAPL $_{\text{CC-clean}}$ trained with 100% of the data. These experiments are run only once and early stopping is based on the validation split of Conceptual Captions.

Pre-trained vision encoder. We ablate the pre-trained vision encoder used to compute image representations. We report results in row (i) of Table 6. We compare two CLIP (Radford et al., 2021) variants, our choice based on the ViT-L/14 backbone and the ViT-B/32 backbone. Indeed, the ViT-L/14 based vision encoder has an average +20% advantage over the ViT-B/32 variant. We hypothesize this improvement is probably due to finer-grained image patches and a bigger model size.

Global vs. grid visual features. Grid features – as opposed to global features – preserve the spatial information in images. This kind of fine-grained information might be useful for complex VL tasks. To measure the impact of grid features, we train a version of MAPL where we use the global image representation from CLIP's multimodal embedding space. Results are reported in row (ii) of Table 6. We observe an average -10% drop in performance, validating our choice of using grid over global visual features.

Mapping network architecture. We ablate the architectural design of our mapping network in rows (iii) and (v) of Table 6. First, we ablate the size of our mapping network in terms of depth and hidden size. We explore three options: Small (2 layers and hidden size of 128), Medium (4 layers and hidden size of 256), and Large (8 layers and hidden size of 512). We see that using a smaller mapping network generally performs slightly worse than the base model. On the other hand, using a larger mapping network improves only in image captioning

	Ablated	Original	Changed	VQAv2	OK-VQA	TextVQA	VizWiz-VQA	CC	COCO	TextCaps	VizWiz-Caps	Overall Δ	
	setting	value	value	4-shot	4-shot	4-shot	4-shot	CIDEr	CIDEr	CIDEr	CIDEr	4-shot	CIDEr
		MAPL		46.39	25.49	9.87	20.02	71.90	54.90	23.30	21.70	0	0
(i)	Vision encoder	CLIP-ViT-L/14	CLIP-ViT-B/32	43.48	24.43	7.85	15.97	59.20	47.00	19.00	15.70	-2.51	-7.73
(ii)	Visual features	Grid	Global	43.75	22.90	8.81	18.20	66.70	49.70	18.40	19.90	-2.03	-4.28
(iii)	Mapping network size	Medium	Small	44.83	26.37	9.68	17.78	68.30	55.50	21.30	20.80	-0.78	-1.48
			Large	45.03	23.92	8.88	19.01	73.40	57.10	24.00	23.10	-1.23	+1.45
(iv)	Output seq. length	32	16	44.18	25.16	9.01	18.15	72.80	56.20	22.50	21.80	-1.32	+0.37
			64	45.22	25.07	10.35	18.89	74.80	58.30	24.30	24.80	-0.56	+2.60
(v)	Learned constant embeddings	Yes	No	40.87	19.31	11.42	16.79	80.52	57.49	30.07	26.16	-3.35	+5.61
(vi)	Data quality Visual features	Clean Grid	Noisy Global	42.80	22.59	6.06	17.33	93.80	42.10	15.60	15.70	-3.25	-1.15

Table 6: Ablation studies. "Overall Δ" refers to the difference (ablated model - base model), averaged across datasets per task.

tasks, while increasing significantly the number of trainable parameters (from 3.4M to 19.5M). We also ablate the output sequence length L_o of our mapping network. Similarly, reducing the output sequence length to 16 yields slightly lower performance overall, and increasing it to 64 only improves in image captioning tasks. In the extreme, we completely remove the learned constant embeddings and output the same sequence length coming from the vision encoder, i.e., $L_o = L_i = 257$. Following the trend, increasing the number of mapped visual embeddings is beneficial for image captioning but hurts VQA performance, while notably reducing training and inference throughput.

Data quality & visual features. This ablation setting aims to be the most similar to Frozen: training on the *full* (noisy) Conceptual Captions dataset while using global visual features. Results are reported in row (vi) of Table 6. The overall performance is worse than that of our base model (-16% on average), but still better than Frozen* $_{\text{CC}}$ on Table 4 (+81% on average). This validates our choice of using grid visual features while training on a subset of cleaner data.

A.6 Additional qualitative results

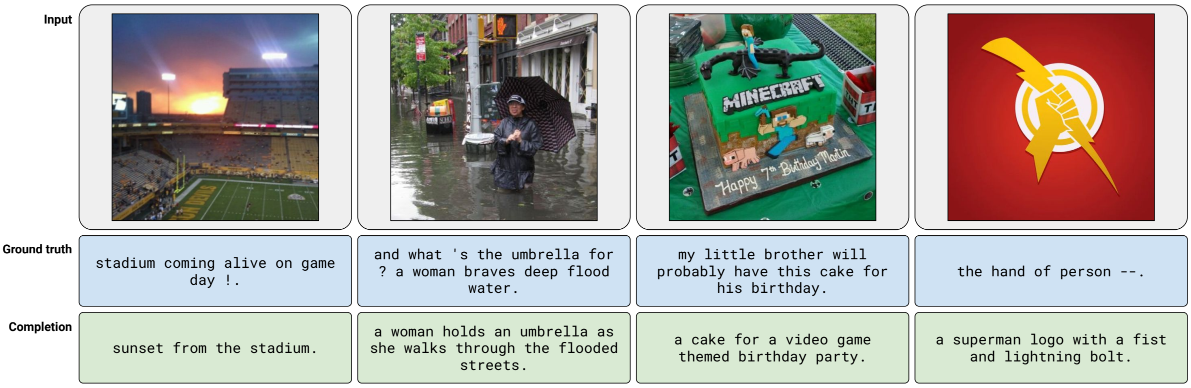
Figures 4-15 show additional qualitative results of MAPL $_{\text{CC-clean}}$ on random samples from different image captioning and VQA datasets. For VQA, in-context learning from 4 shots is performed.

A.7 Interpretability of visual embeddings

Using MAPL $_{CC-clean}$, we extract mapped visual embeddings (after the mapping network) for ∼30 images from the COCO Karpathy-test set, and compute the nearest token embeddings (from the LM's vocabulary) using cosine similarity. We rarely found the top-5 nearest tokens correspond to concepts present in the image, suggesting these embeddings are not interpretable. We hypothesize this is perhaps because they carry a combination of task-inducing and image-specific information, also pointed out by Mokady et al. (2021). We further cluster the mapped visual embeddings with K-means, and observe that each cluster often represents some visual concept (e.g., animals, food, sports). This means the mapped visual embeddings retain visual information from the vision encoder, which we also verify with MAPL's performance on VL benchmarks.

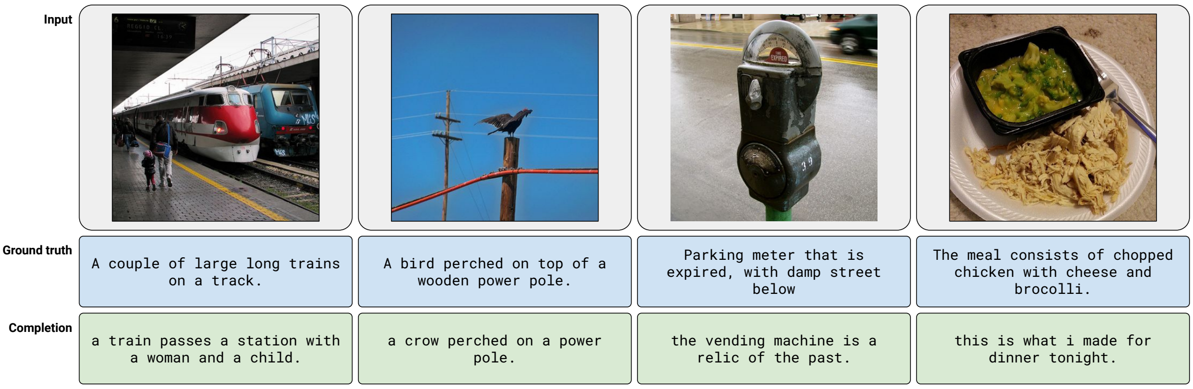
A.8 Analysis of VQA answer distributions

In this section, we show the distribution of answers for selected VQAv2 question types. We compare MAPL with several baselines of our model to get insights into how the model's predictions change when training on increasing multimodal data. For the text-only baseline, we only provide the question text to the LM. This is different from the previously introduced blind baseline (Sec. 4.1), where a blacked-out image is also provided. In particular, we compare the predicted answer distribution of MAPL $_{\text{COCO}}$ evaluated on on zero- and few-shot VQA with the aforementioned baselines and the ground truth. Overall, we observe the predicted answer distribution gets closer to the ground truth answer distribution (Figure 21) as more information from the image-question pair is provided to the model. We notice a considerable shift in the answer distribution from the text-only baseline (Figure 16) to the blind baseline, which demonstrates the impact caused by the captions alone. Moreover, we see the predicted answer distribution of MAPL zero-shot is closer to the ground truth answer distribution than that of the blind baseline (Figure 17), which indicates that MAPL is leveraging the additional information from the visual input. For instance, we observe MAPL's predicted answer distribution for the "what color" question type (second column) looks more similar to the ground truth distribution compared to the text-only and blind baselines. Finally, when performing in-context learning from four shots (Figure 19), we see the answer distribution gets even closer to the ground truth distribution. However, we do not observe much difference in answer distribution when increasing the number of shots from four to eight (Figure 20).



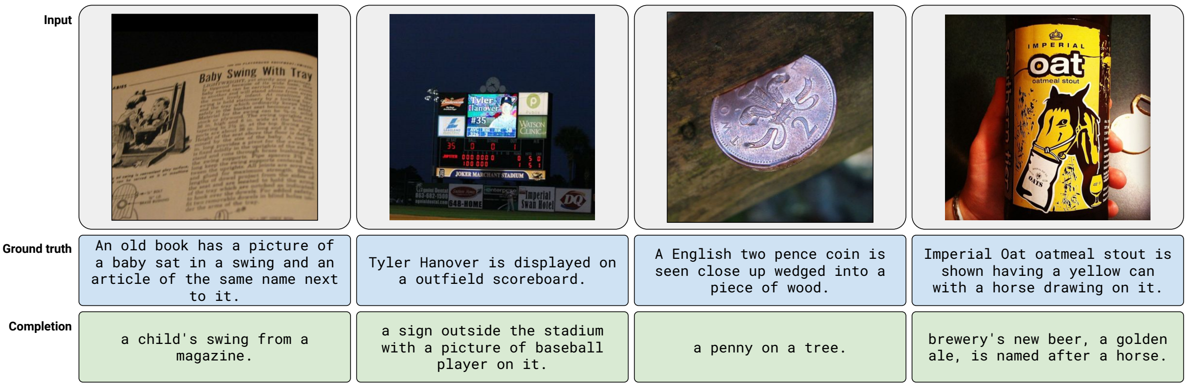
Ground truth	stadium coming alive on game day !.	and what 's the umbrella for ? a woman braves deep flood water.	my little brother will probably have this cake for his birthday.	the hand of person --.
Completion	sunset from the stadium.	a woman holds an umbrella as she walks through the flooded streets.	a cake for a video game themed birthday party.	a superman logo with a fist and lightning bolt.

Figure 4: MAPL's image captioning on Conceptual Captions.



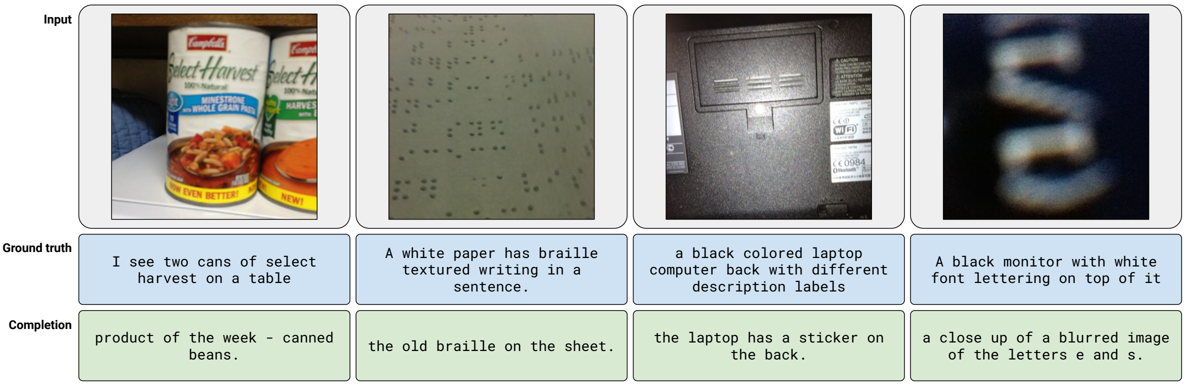
Ground truth	A couple of large long trains on a track.	A bird perched on top of a wooden power pole.	Parking meter that is expired, with damp street below	The meal consists of chopped chicken with cheese and brocolli.
Completion	a train passes a station with a woman and a child.	a crow perched on a power pole.	the vending machine is a relic of the past.	this is what i made for dinner tonight.

Figure 5: MAPL's image captioning on COCO Captions.



Ground truth	An old book has a picture of a baby sat in a swing and an article of the same name next to it.	Tyler Hanover is displayed on a outfield scoreboard.	A English two pence coin is seen close up wedged into a piece of wood.	Imperial Oat oatmeal stout is shown having a yellow can with a horse drawing on it.
Completion	a child's swing from a magazine.	a sign outside the stadium with a picture of baseball player on it.	a penny on a tree.	brewery's new beer, a golden ale, is named after a horse.

Figure 6: MAPL's image captioning on TextCaps.



Ground truth	I see two cans of select harvest on a table	A white paper has braille textured writing in a sentence.	a black colored laptop computer back with different description labels	A black monitor with white font lettering on top of it
Completion	product of the week - canned beans.	the old braille on the sheet.	the laptop has a sticker on the back.	a close up of a blurred image of the letters e and s.

Figure 7: MAPL's image captioning on VizWiz Captions.

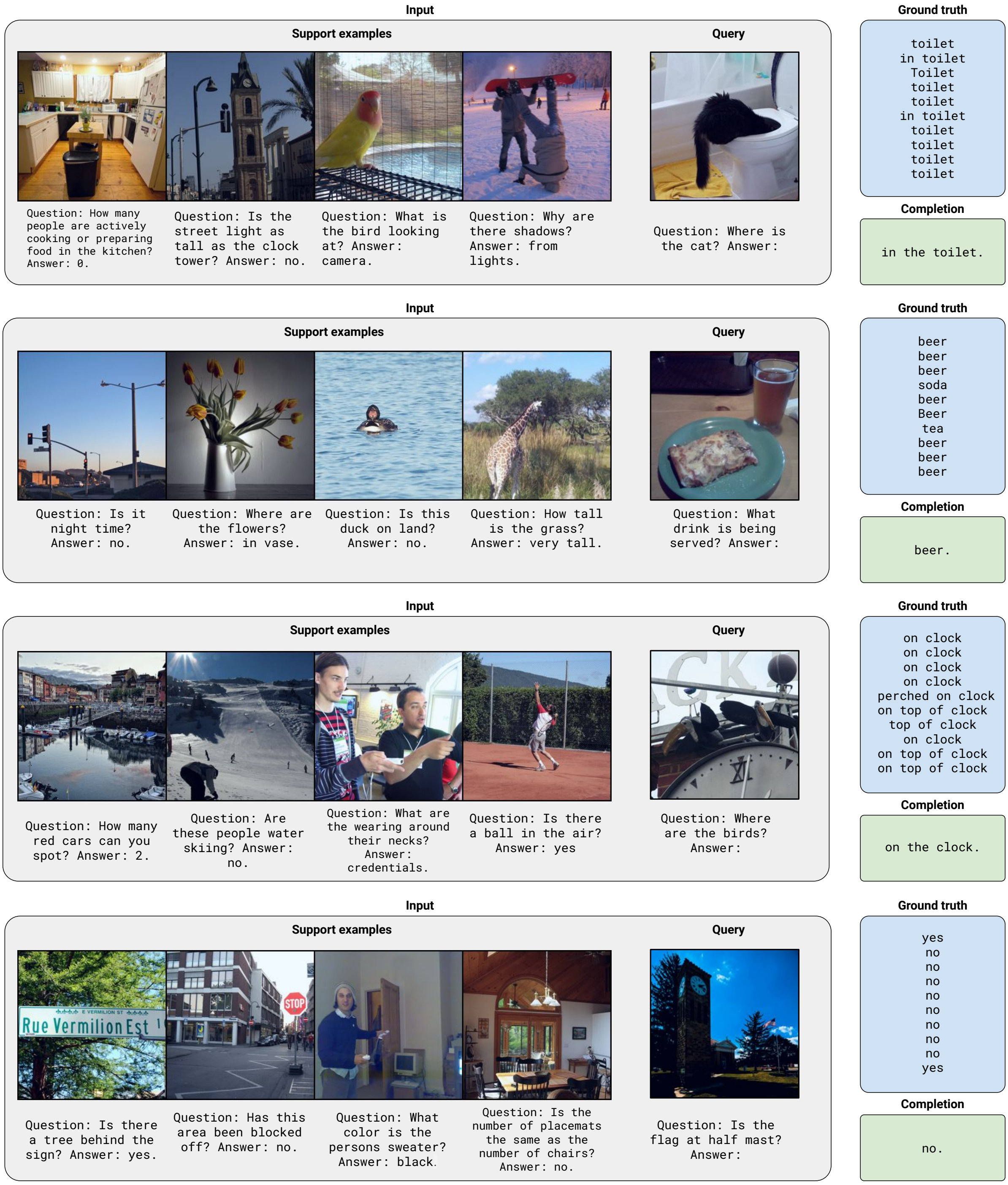
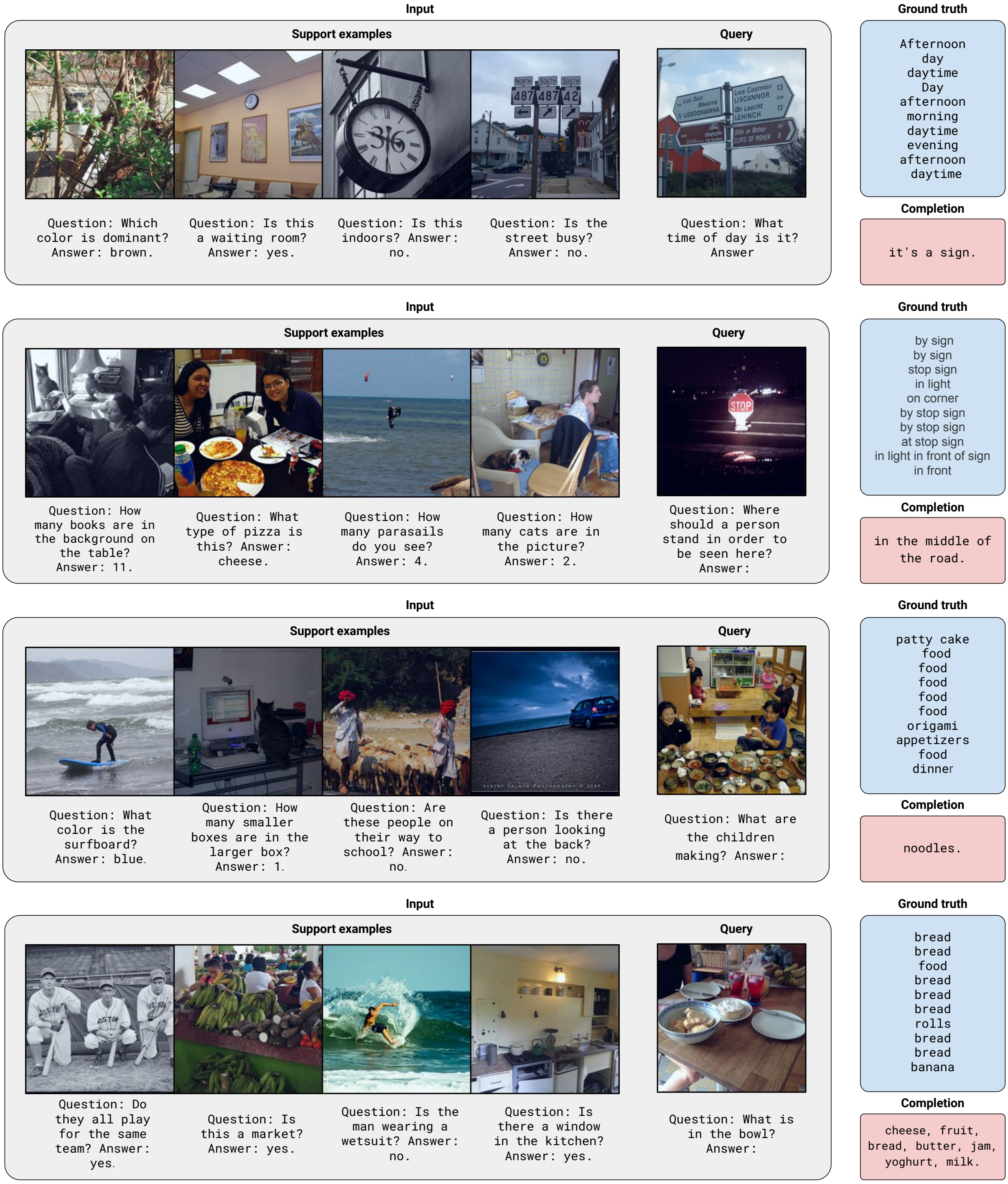

Input
Support examples
Query
Ground truth
Question: How many people are actively cooking or preparing food in the kitchen? Answer: 0.
Question: Is the street light as tall as the clock tower? Answer: no.
Question: What is the bird looking at? Answer: camera.
Question: Why are there shadows? Answer: from lights.
Question: Where is the cat? Answer:
toilet
in toilet
Toilet
toilet
toilet
in toilet
toilet
toilet
toilet
toilet
Completion
in the toilet.
Input
Support examples
Query
Ground truth
Question: Is it night time? Answer: no.
Question: Where are the flowers? Answer: in vase.
Question: Is this duck on land? Answer: no.
Question: How tall is the grass? Answer: very tall.
Question: What drink is being served? Answer:
beer
beer
beer
soda
beer
Beer
tea
beer
beer
beer
Completion
beer.
Input
Support examples
Query
Ground truth
Question: How many red cars can you spot? Answer: 2.
Question: Are these people water skiing? Answer: no.
Question: What are the wearing around their necks? Answer: credentials.
Question: Is there a ball in the air? Answer: yes
Question: Where are the birds? Answer:
on clock
on clock
on clock
on clock
perched on clock
on top of clock
top of clock
on clock
on top of clock
on top of clock
Completion
on the clock.
Input
Support examples
Query
Ground truth
Rue Vermilion Est
E VERMILION ST
STOP
Question: Is there a tree behind the sign? Answer: yes.
Question: Has this area been blocked off? Answer: no.
Question: What color is the persons sweater? Answer: black.
Question: Is the number of placemats the same as the number of chairs? Answer: no.
Question: Is the flag at half mast? Answer:
yes
no
no
no
no
no
no
no
no
yes
Completion
no.

Figure 8: MAPL's 4-shot VQA on VQAv2, success cases.



Question: Which color is dominant? Answer: brown.

Question: Is this a waiting room? Answer: yes.

Question: Is this indoors? Answer: no.

Question: Is the street busy? Answer: no.

Question: What time of day is it? Answer

Afternoon
day
daytime
Day
afternoon
morning
daytime
evening
afternoon
daytime

it's a sign.

Question: How many books are in the background on the table? Answer: 11.

Question: What type of pizza is this? Answer: cheese.

Question: How many parasails do you see? Answer: 4.

Question: How many cats are in the picture? Answer: 2.

Question: Where should a person stand in order to be seen here? Answer:

by sign
by sign
stop sign
in light
on corner
by stop sign
by stop sign
at stop sign
in light in front of sign
in front

in the middle of the road.

Question: What color is the surfboard? Answer: blue.

Question: How many smaller boxes are in the larger box? Answer: 1.

Question: Are these people on their way to school? Answer: no.

Question: Is there a person looking at the back? Answer: no.

Question: What are the children making? Answer:

patty cake
food
food
food
food
food
origami
appetizers
food
dinner

noodles.

Question: Do they all play for the same team? Answer: yes.

Question: Is this a market? Answer: yes.

Question: Is the man wearing a wetsuit? Answer: no.

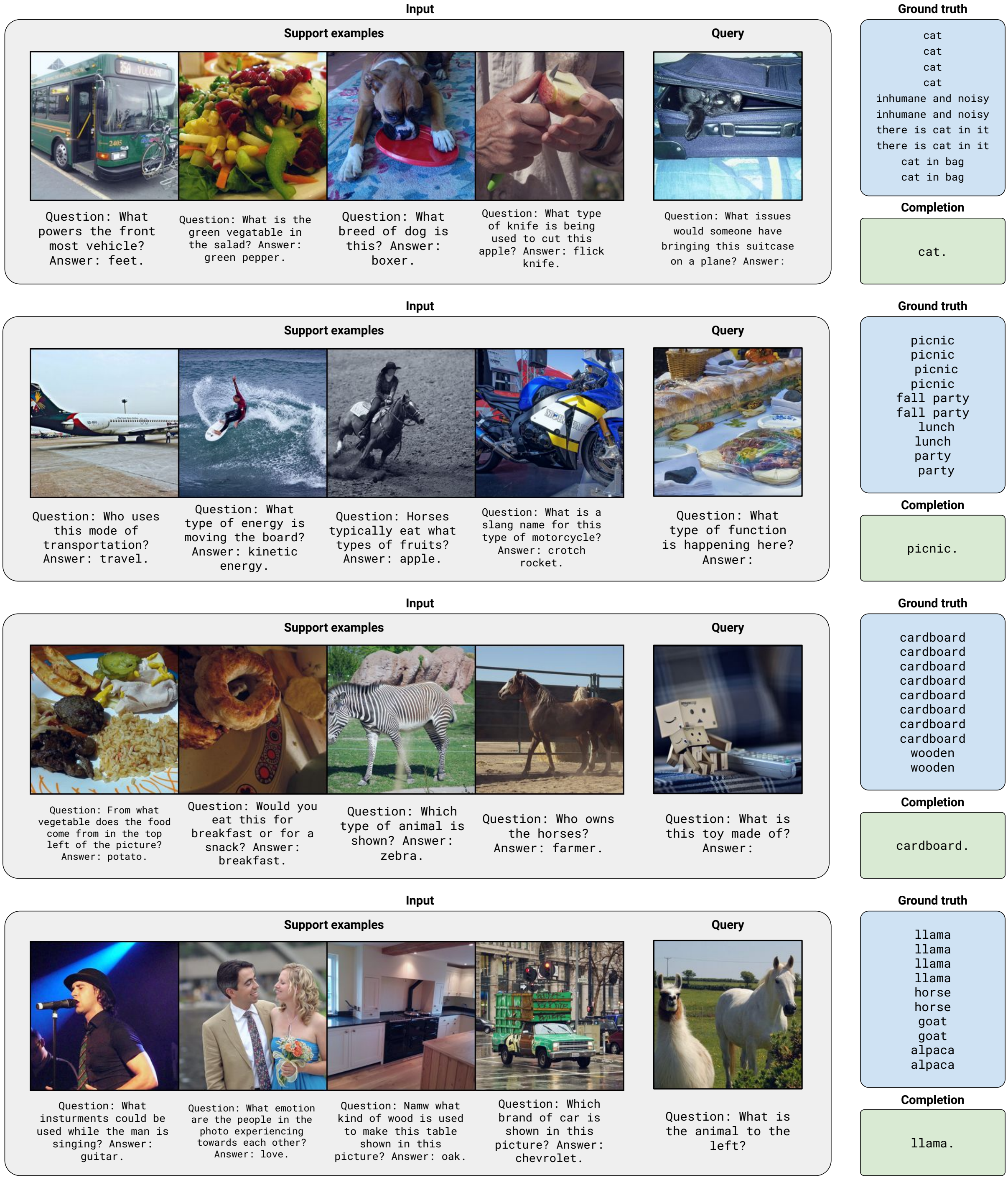
Question: Is there a window in the kitchen? Answer: yes.

Question: What is in the bowl? Answer:

bread
bread
food
bread
bread
bread
rolls
bread
bread
banana

cheese, fruit, bread, butter, jam, yoghurt, milk.

Figure 9: MAPL's 4-shot VQA on VQAv2, failure cases.



Question: What powers the front most vehicle? Answer: feet.

Question: What is the green vegatable in the salad? Answer: green pepper.

Question: What breed of dog is this? Answer: boxer.

Question: What type of knife is being used to cut this apple? Answer: flick knife.

Question: What issues would someone have bringing this suitcase on a plane? Answer:

cat
cat
cat
cat
inhumane and noisy
inhumane and noisy
there is cat in it
there is cat in it
cat in bag
cat in bag

cat.

Question: Who uses this mode of transportation? Answer: travel.

Question: What type of energy is moving the board? Answer: kinetic energy.

Question: Horses typically eat what types of fruits? Answer: apple.

Question: What is a slang name for this type of motorcycle? Answer: crotch rocket.

Question: What type of function is happening here? Answer:

picnic
picnic
picnic
picnic
fall party
fall party
lunch
lunch
party
party

picnic.

Question: From what vegetable does the food come from in the top left of the picture? Answer: potato.

Question: Would you eat this for breakfast or for a snack? Answer: breakfast.

Question: Which type of animal is shown? Answer: zebra.

Question: Who owns the horses? Answer: farmer.

Question: What is this toy made of? Answer:

cardboard
cardboard
cardboard
cardboard
cardboard
cardboard
cardboard
cardboard
wooden
wooden

cardboard.

Question: What insturments could be used while the man is singing? Answer: guitar.

Question: What emotion are the people in the photo experiencing towards each other? Answer: love.

Question: Namw what kind of wood is used to make this table shown in this picture? Answer: oak.

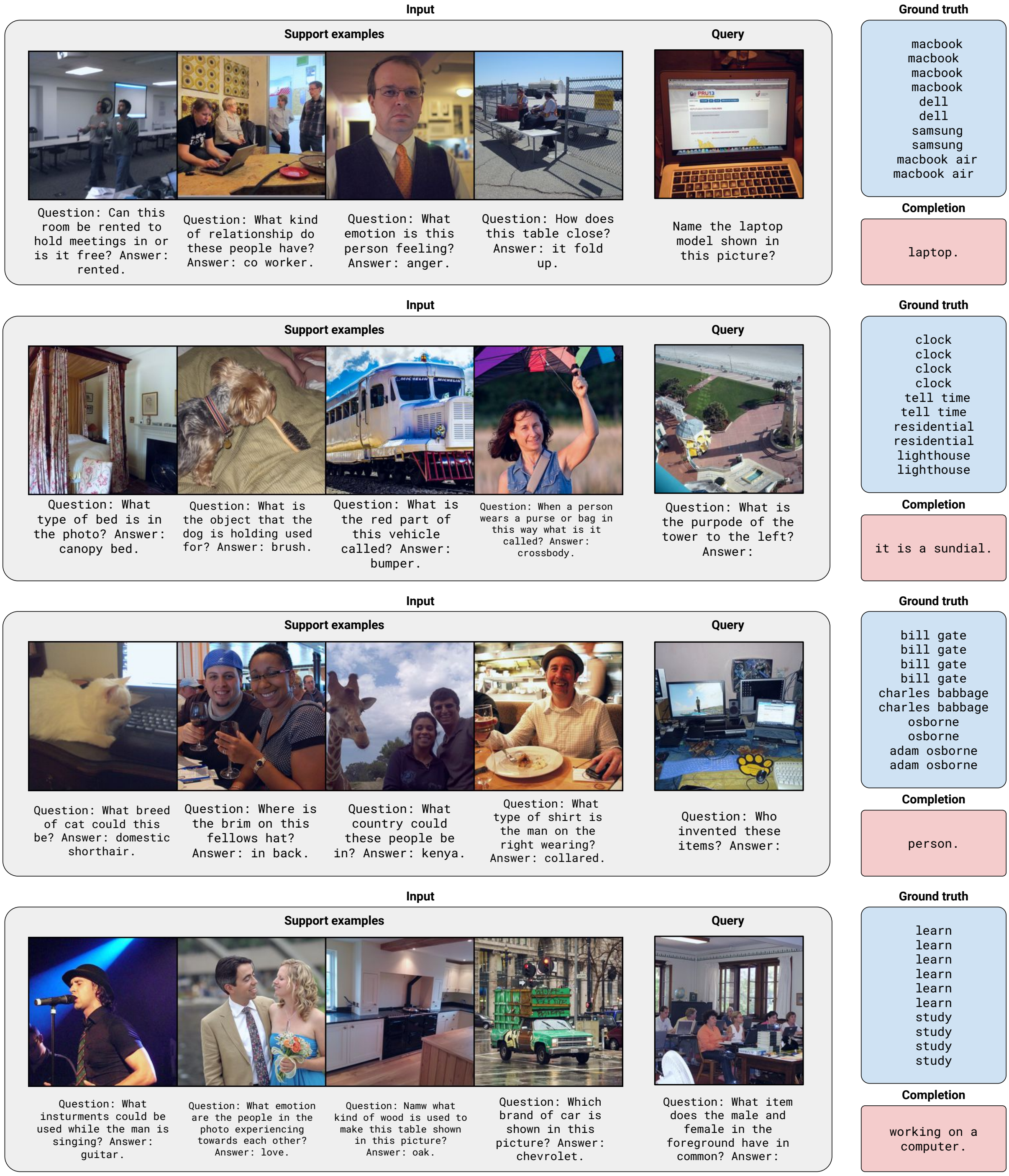
Question: Which brand of car is shown in this picture? Answer: chevrolet.

Question: What is the animal to the left?

llama
llama
llama
llama
horse
horse
goat
goat
alpaca
alpaca

llama.

Figure 10: MAPL's 4-shot VQA on OK-VQA, success cases.



Question: Can this room be rented to hold meetings in or is it free? Answer: rented.

Question: What kind of relationship do these people have? Answer: co worker.

Question: What emotion is this person feeling? Answer: anger.

Question: How does this table close? Answer: it fold up.

Name the laptop model shown in this picture?

Ground truth: macbook macbook macbook macbook dell dell samsung samsung macbook air macbook air

Completion: laptop.

Question: What type of bed is in the photo? Answer: canopy bed.

Question: What is the object that the dog is holding used for? Answer: brush.

Question: What is the red part of this vehicle called? Answer: bumper.

Question: When a person wears a purse or bag in this way what is it called? Answer: crossbody.

Question: What is the purpode of the tower to the left? Answer:

Ground truth: clock clock clock clock tell time tell time residential residential lighthouse lighthouse

Completion: it is a sundial.

Question: What breed of cat could this be? Answer: domestic shorthair.

Question: Where is the brim on this fellows hat? Answer: in back.

Question: What country could these people be in? Answer: kenya.

Question: What type of shirt is the man on the right wearing? Answer: collared.

Question: Who invented these items? Answer:

Ground truth: bill gate bill gate bill gate bill gate charles babbage charles babbage osborne osborne adam osborne adam osborne

Completion: person.

Question: What insturments could be used while the man is singing? Answer: guitar.

Question: What emotion are the people in the photo experiencing towards each other? Answer: love.

Question: Namw what kind of wood is used to make this table shown in this picture? Answer: oak.

Question: Which brand of car is shown in this picture? Answer: chevrolet.

Question: What item does the male and female in the foreground have in common? Answer:

Ground truth: learn learn learn learn learn learn study study study study

Completion: working on a computer.

Figure 11: MAPL's 4-shot VQA on OK-VQA, failure cases.

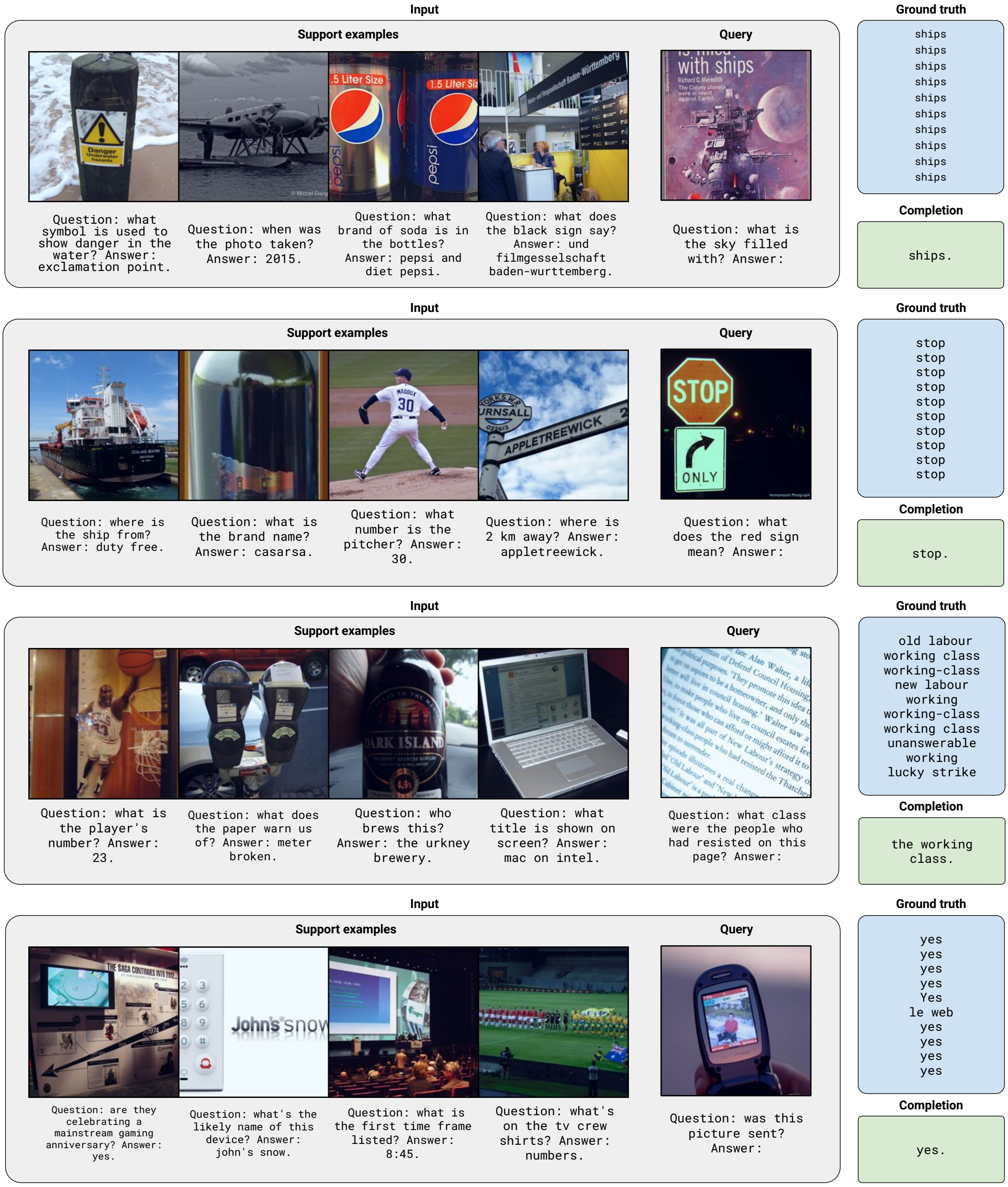
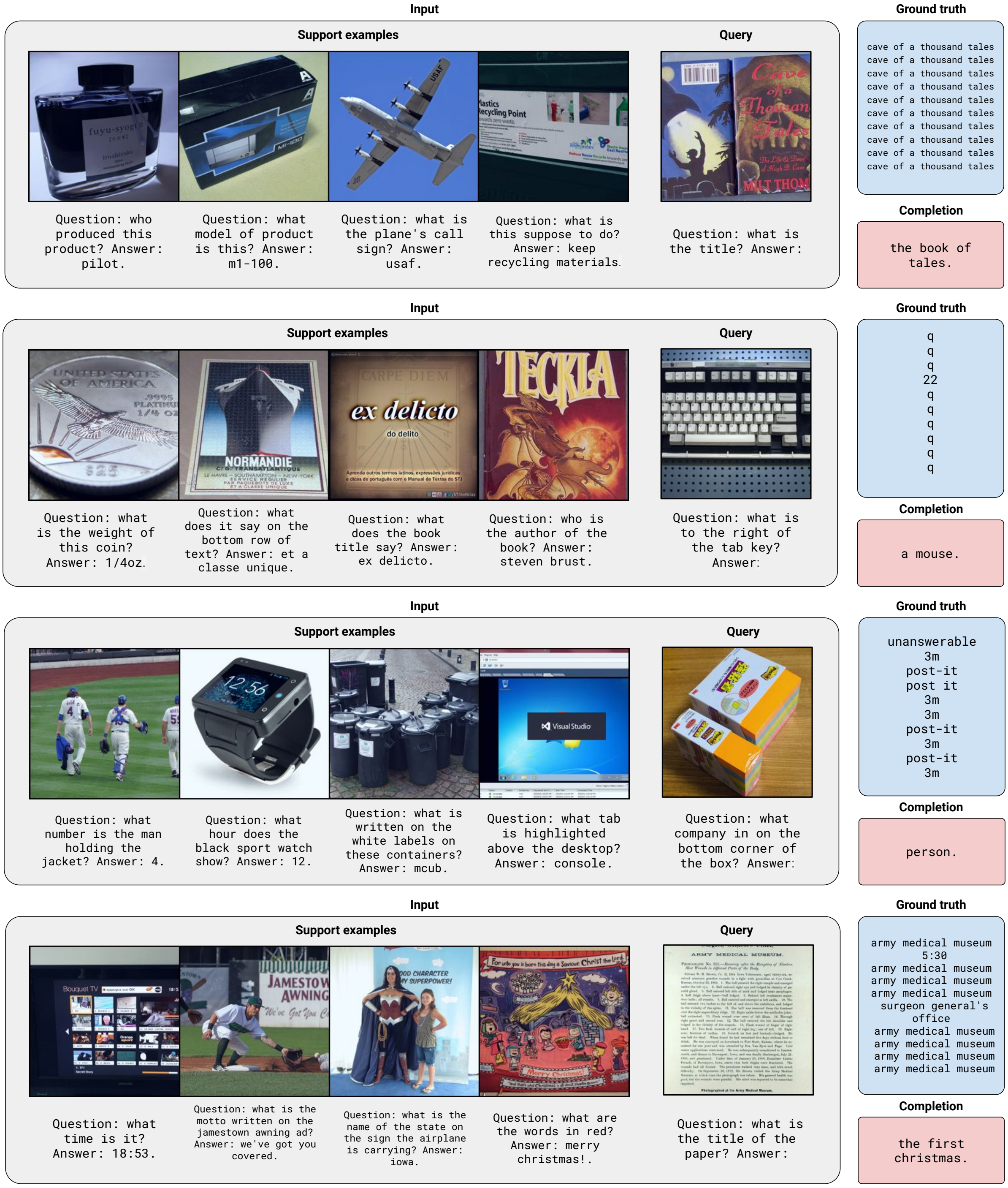

Input
Support examples
Query
Question: what symbol is used to show danger in the water? Answer: exclamation point.
Question: when was the photo taken? Answer: 2015.
Question: what brand of soda is in the bottles? Answer: pepsi and diet pepsi.
Question: what does the black sign say? Answer: und filmgesselschaft baden-wurttemberg.
Question: what is the sky filled with? Answer:
Ground truth
ships
ships
ships
ships
ships
ships
ships
ships
ships
ships
Completion
ships.
Input
Support examples
Query
Question: where is the ship from? Answer: duty free.
Question: what is the brand name? Answer: casarsa.
Question: what number is the pitcher? Answer: 30.
Question: where is 2 km away? Answer: appletreewick.
Question: what does the red sign mean? Answer:
Ground truth
stop
stop
stop
stop
stop
stop
stop
stop
stop
stop
Completion
stop.
Input
Support examples
Query
Question: what is the player's number? Answer: 23.
Question: what does the paper warn us of? Answer: meter broken.
Question: who brews this? Answer: the urkney brewery.
Question: what title is shown on screen? Answer: mac on intel.
Question: what class were the people who had resisted on this page? Answer:
Ground truth
old labour
working class
working-class
new labour
working
working-class
working class
unanswerable
working
lucky strike
Completion
the working class.
Input
Support examples
Query
Question: are they celebrating a mainstream gaming anniversary? Answer: yes.
Question: what's the likely name of this device? Answer: john's snow.
Question: what is the first time frame listed? Answer: 8:45.
Question: what's on the tv crew shirts? Answer: numbers.
Question: was this picture sent? Answer:
Ground truth
yes
yes
yes
yes
Yes
le web
yes
yes
yes
yes
Completion
yes.

Figure 12: MAPL's 4-shot VQA on TextVQA, success cases.



Question: who produced this product? Answer: pilot.

Question: what model of product is this? Answer: m1-100.

Question: what is the plane's call sign? Answer: usaf.

Question: what is this suppose to do? Answer: keep recycling materials.

Question: what is the title? Answer:

cave of a thousand tales
cave of a thousand tales
cave of a thousand tales
cave of a thousand tales
cave of a thousand tales
cave of a thousand tales
cave of a thousand tales
cave of a thousand tales
cave of a thousand tales
cave of a thousand tales

the book of tales.

Question: what is the weight of this coin? Answer: 1/4oz.

Question: what does it say on the bottom row of text? Answer: et a classe unique.

Question: what does the book title say? Answer: ex delicto.

Question: who is the author of the book? Answer: steven brust.

Question: what is to the right of the tab key? Answer:

q
q
q
22
q
q
q
q
q
q

a mouse.

Question: what number is the man holding the jacket? Answer: 4.

Question: what hour does the black sport watch show? Answer: 12.

Question: what is written on the white labels on these containers? Answer: mcub.

Question: what tab is highlighted above the desktop? Answer: console.

Question: what company in on the bottom corner of the box? Answer:

unanswerable
3m
post-it
post it
3m
3m
post-it
3m
post-it
3m

person.

Question: what time is it? Answer: 18:53.

Question: what is the motto written on the jamestown awning ad? Answer: we've got you covered.

Question: what is the name of the state on the sign the airplane is carrying? Answer: iowa.

Question: what are the words in red? Answer: merry christmas!.

Question: what is the title of the paper? Answer:

army medical museum
5:30
army medical museum
army medical museum
army medical museum
surgeon general's office
army medical museum
army medical museum
army medical museum
army medical museum

the first christmas.

Figure 13: MAPL's 4-shot VQA on TextVQA, failure cases.

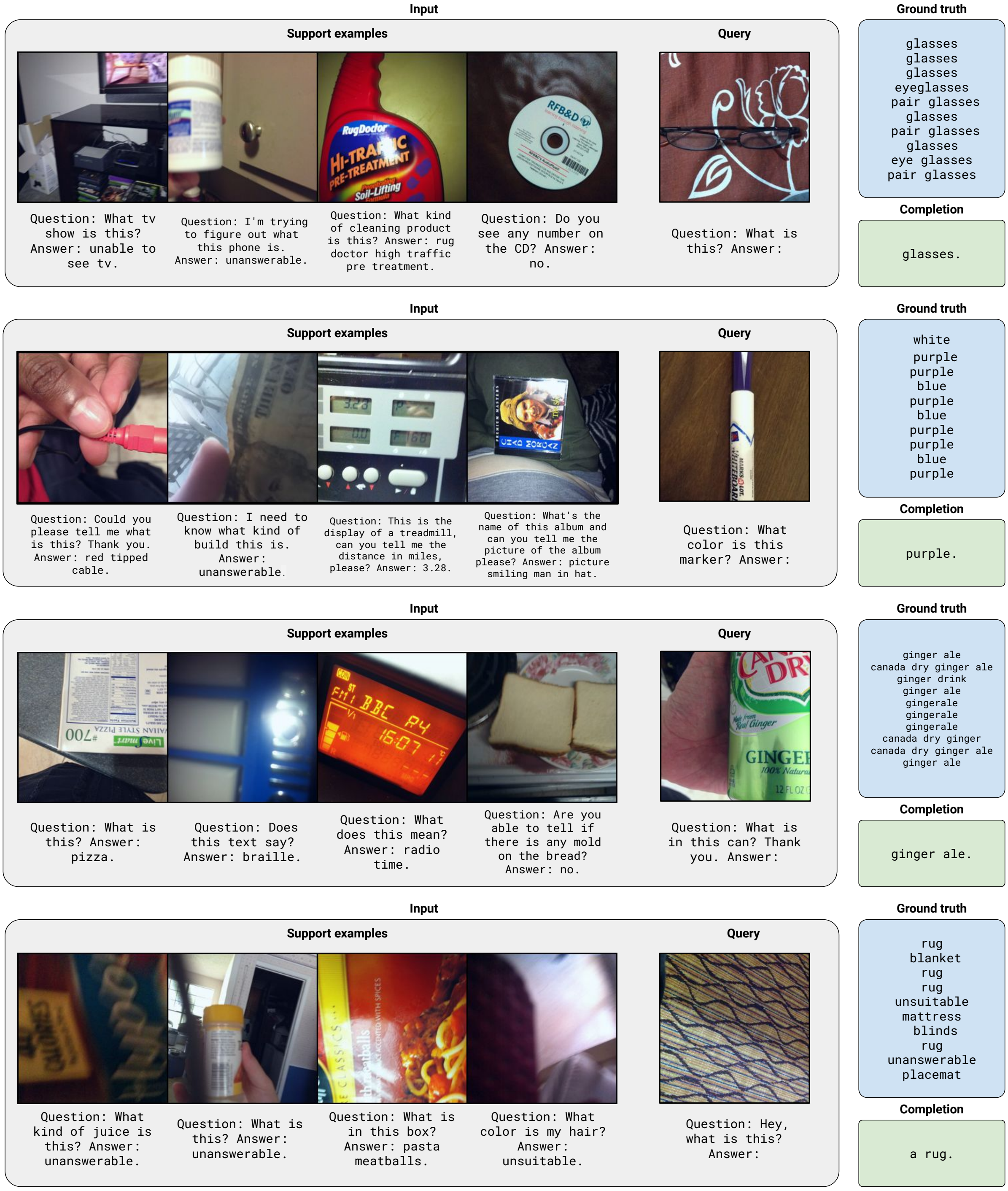
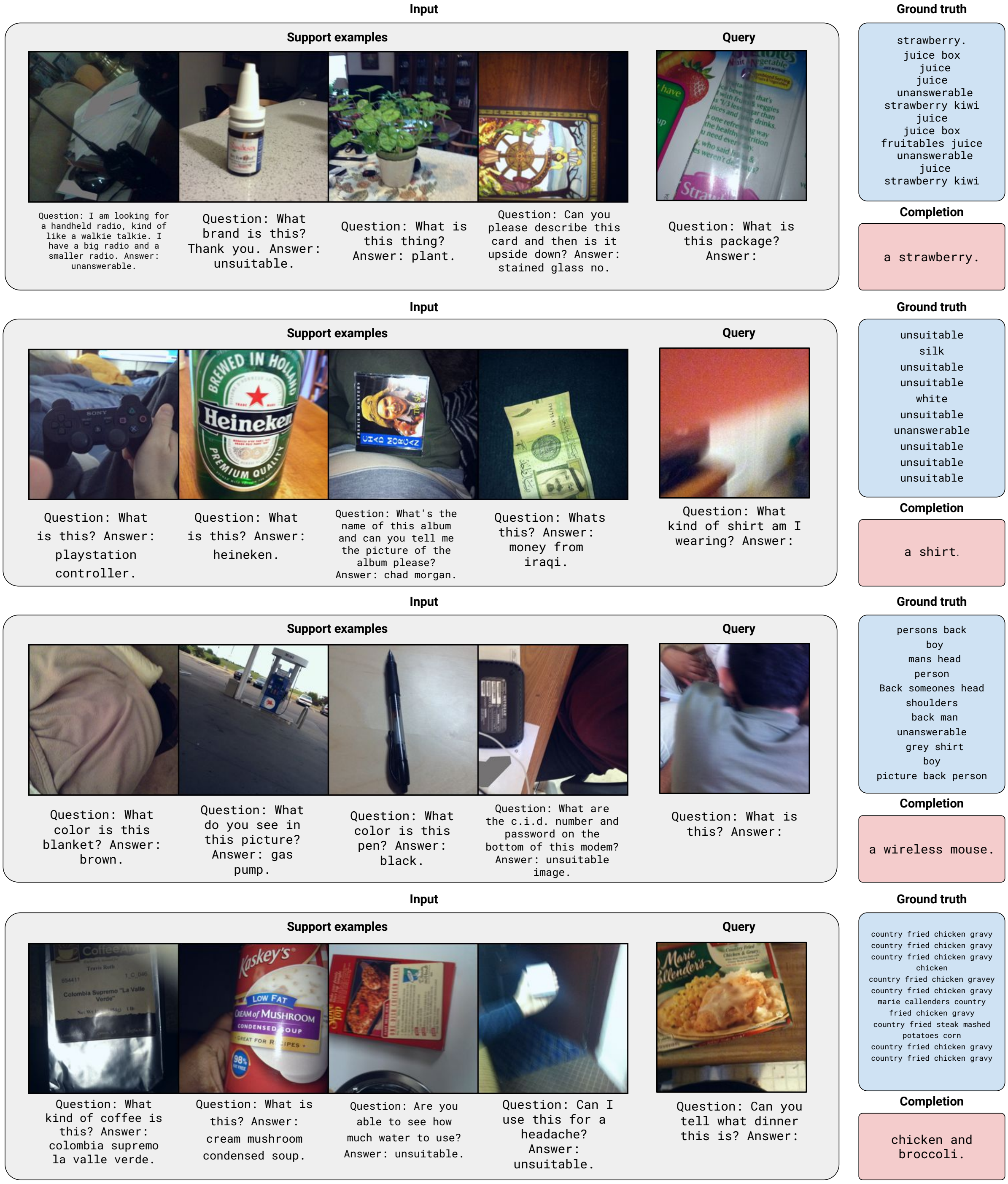

Input
Support examples
Query
Ground truth
Question: What tv show is this? Answer: unable to see tv.
Question: I'm trying to figure out what this phone is. Answer: unanswerable.
Question: What kind of cleaning product is this? Answer: rug doctor high traffic pre treatment.
Question: Do you see any number on the CD? Answer: no.
Question: What is this? Answer:
glasses
glasses
glasses
eyeglasses
pair glasses
glasses
pair glasses
glasses
eye glasses
pair glasses
Completion
glasses.
Input
Support examples
Query
Ground truth
Question: Could you please tell me what is this? Thank you. Answer: red tipped cable.
Question: I need to know what kind of build this is. Answer: unanswerable.
Question: This is the display of a treadmill, can you tell me the distance in miles, please? Answer: 3.28.
Question: What's the name of this album and can you tell me the picture of the album please? Answer: picture smiling man in hat.
Question: What color is this marker? Answer:
white
purple
purple
blue
purple
blue
purple
purple
blue
purple
Completion
purple.
Input
Support examples
Query
Ground truth
Question: What is this? Answer: pizza.
Question: Does this text say? Answer: braille.
Question: What does this mean? Answer: radio time.
Question: Are you able to tell if there is any mold on the bread? Answer: no.
Question: What is in this can? Thank you. Answer:
ginger ale
canada dry ginger ale
ginger drink
ginger ale
gingerale
gingerale
gingerale
canada dry ginger
canada dry ginger ale
ginger ale
Completion
ginger ale.
Input
Support examples
Query
Ground truth
Question: What kind of juice is this? Answer: unanswerable.
Question: What is this? Answer: unanswerable.
Question: What is in this box? Answer: pasta meatballs.
Question: What color is my hair? Answer: unsuitable.
Question: Hey, what is this? Answer:
rug
blanket
rug
rug
unsuitable
mattress
blinds
rug
unanswerable
placemat
Completion
a rug.

Figure 14: MAPL's 4-shot VQA on VizWiz-VQA, success cases.



Question: I am looking for a handheld radio, kind of like a walkie talkie. I have a big radio and a smaller radio. Answer: unanswerable.

Question: What brand is this? Thank you. Answer: unsuitable.

Question: What is this thing? Answer: plant.

Question: Can you please describe this card and then is it upside down? Answer: stained glass no.

Question: What is this package? Answer:

strawberry.
juice box
juice
juice
unanswerable
strawberry kiwi
juice
juice box
fruitables juice
unanswerable
juice
strawberry kiwi

a strawberry.

Question: What is this? Answer: playstation controller.

Question: What is this? Answer: heineken.

Question: What's the name of this album and can you tell me the picture of the album please? Answer: chad morgan.

Question: Whats this? Answer: money from iraqi.

Question: What kind of shirt am I wearing? Answer:

unsuitable
silk
unsuitable
unsuitable
white
unsuitable
unanswerable
unsuitable
unsuitable
unsuitable

a shirt.

Question: What color is this blanket? Answer: brown.

Question: What do you see in this picture? Answer: gas pump.

Question: What color is this pen? Answer: black.

Question: What are the c.i.d. number and password on the bottom of this modem? Answer: unsuitable image.

Question: What is this? Answer:

persons back
boy
mans head
person
Back someones head
shoulders
back man
unanswerable
grey shirt
boy
picture back person

a wireless mouse.

Question: What kind of coffee is this? Answer: colombia supremo la valle verde.

Question: What is this? Answer: cream mushroom condensed soup.

Question: Are you able to see how much water to use? Answer: unsuitable.

Question: Can I use this for a headache? Answer: unsuitable.

Question: Can you tell what dinner this is? Answer:

country fried chicken gravy
country fried chicken gravy
country fried chicken gravy chicken
country fried chicken gravey
country fried chicken gravy
marie callenders country fried chicken gravy
country fried steak mashed potatoes corn
country fried chicken gravy
country fried chicken gravy

chicken and broccoli.

Figure 15: MAPL's 4-shot VQA on VizWiz-VQA, failure cases.

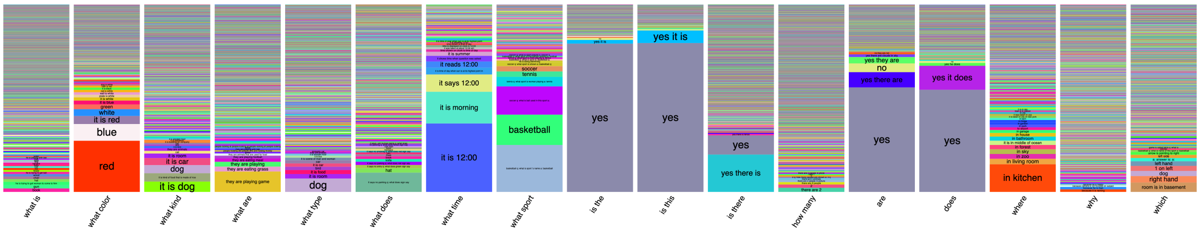


Figure 16: Predicted answer distributions for selected VQAv2 question types with the text-only baseline.

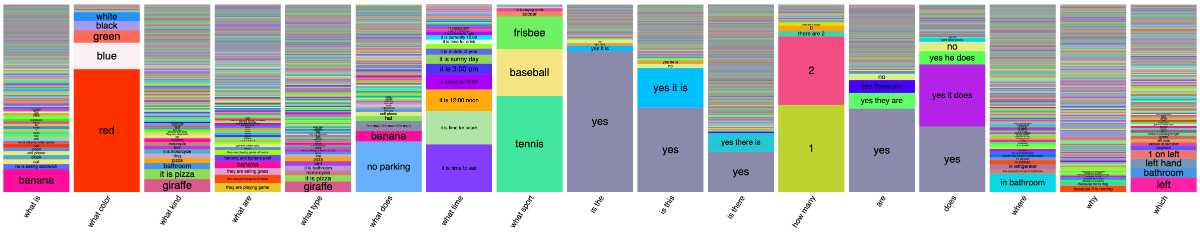


Figure 17: Predicted answer distributions for selected VQAv2 question types with the blind baseline.

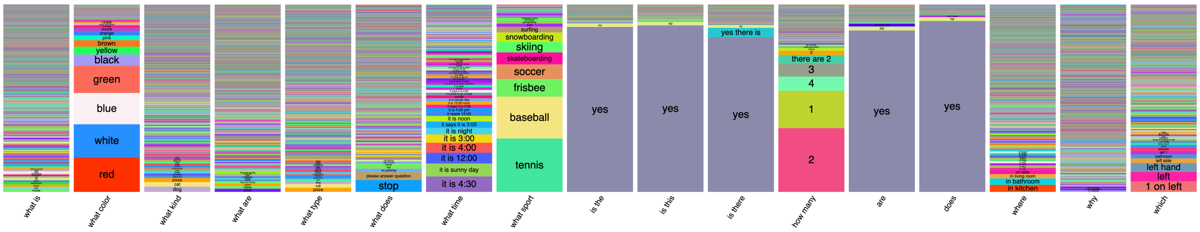


Figure 18: Predicted answer distributions for selected VQAv2 question types with MAPL 0-shot.

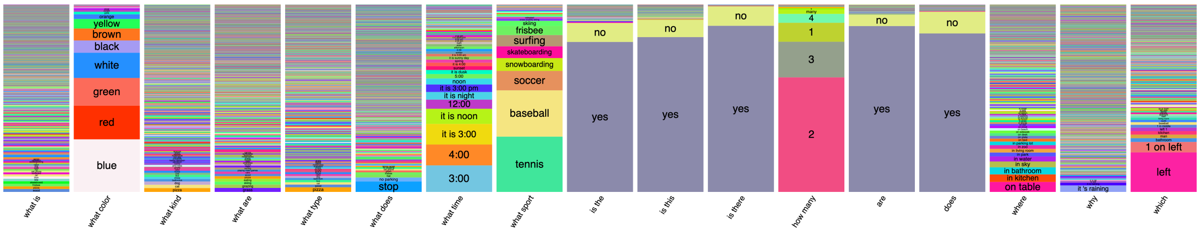


Figure 19: Predicted answer distributions for selected VQAv2 question types with MAPL 4-shot.

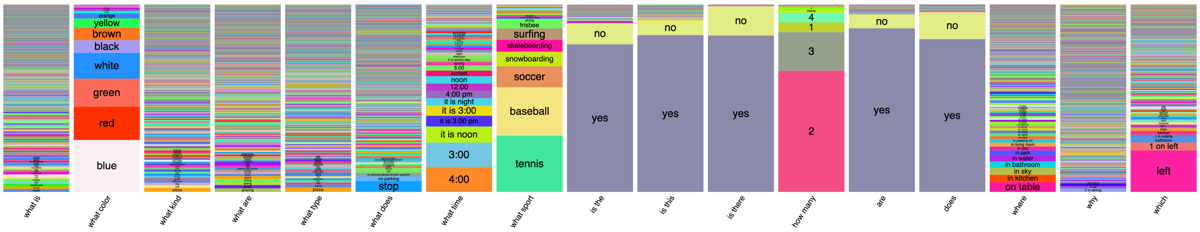


Figure 20: Predicted answer distributions for selected VQAv2 question types with MAPL 8-shot.

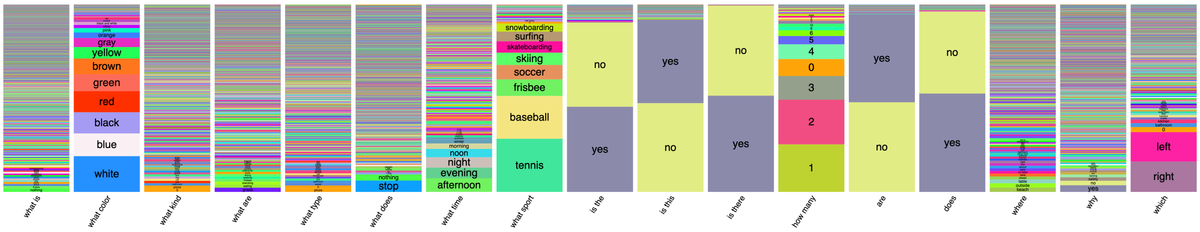


Figure 21: Ground truth answer distributions for selected VQAv2 question types.