🦩 Flamingo: a Visual Language Model for Few-Shot Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

Building models that can be rapidly adapted to novel tasks using only a handful of annotated examples is an open challenge for multimodal machine learning research. We introduce Flamingo, a family of Visual Language Models (VLM) with this ability. We propose key architectural innovations to: (i) bridge powerful pretrained vision-only and language-only models, (ii) handle sequences of arbitrarily interleaved visual and textual data, and (iii) seamlessly ingest images or videos as inputs. Thanks to their flexibility, Flamingo models can be trained on large-scale multimodal web corpora containing arbitrarily interleaved text and images, which is key to endow them with in-context few-shot learning capabilities. We perform a thorough evaluation of our models, exploring and measuring their ability to rapidly adapt to a variety of image and video tasks. These include open-ended tasks such as visual question-answering, where the model is prompted with a question which it has to answer; captioning tasks, which evaluate the ability to describe a scene or an event; and close-ended tasks such as multiple-choice visual question-answering. For tasks lying anywhere on this spectrum, a *single* Flamingo model can achieve a new state of the art with few-shot learning, simply by prompting the model with task-specific examples. On numerous benchmarks, *Flamingo* outperforms models fine-tuned on thousands of times more task-specific data.

1 Introduction

One key aspect of intelligence is the ability to quickly learn to perform a new task given a short instruction [32, 67]. While initial progress has been made towards a similar capability in computer vision, the most widely used paradigm still consists of first pretraining on a large amount of supervised data, before fine-tuning the model on the task of interest [63, 114, 139]. However, successful fine-tuning often requires many thousands of annotated data points. In addition, it often requires careful per-task hyperparameter tuning and is also resource intensive. Recently, multimodal vision-language models trained with a contrastive objective [48, 82] have enabled zero-shot adaptation to novel tasks, without the need for fine-tuning. However, because these models simply provide a similarity score between a text and an image, they can only address limited use cases such as classification, where a finite set of outcomes is provided beforehand. They crucially lack the ability to generate language, which makes them less suitable to more open-ended tasks such as captioning or visual question-answering. Others have explored visually-conditioned language generation [17, 110, 115, 120, 128] but have not yet shown good performance in low-data regimes.

We introduce *Flamingo*, a Visual Language Model (VLM) that sets a new state of the art in few-shot learning on a wide range of open-ended vision and language tasks, simply by being prompted with a few input/output examples, as illustrated in Figure 1. Of the 16 tasks we consider, *Flamingo* also surpasses the fine-tuned state of the art on 6 tasks, despite using orders of magnitude less task-specific

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Figure 1: **Examples of inputs and outputs obtained from *Flamingo*-80B.** *Flamingo* can rapidly adapt to various image/video understanding tasks with few-shot prompting (top). Out of the box, *Flamingo* is also capable of multi-image visual dialogue (bottom). More examples in Appendix C.

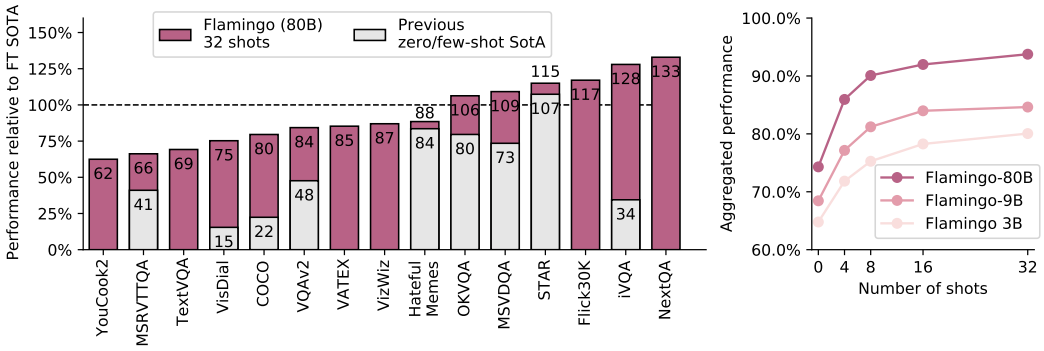

Figure 2: **Flamingo results overview.** *Left*: Our largest model, dubbed *Flamingo*, outperforms state-of-the-art fine-tuned models on 6 of the 16 tasks we consider with no fine-tuning. For the 9 tasks with published few-shot results, *Flamingo* sets the new few-shot state of the art. *Note:* We omit RareAct, our 16th benchmark, as it is a zero-shot benchmark with no available fine-tuned results to compare to. *Right*: Flamingo performance improves with model size and number of shots.

training data (see Figure 2). To achieve this, Flamingo takes inspiration from recent work on large language models (LMs) which are good few-shot learners [11, 18, 41, 83]. A single large LM can achieve strong performance on many tasks using only its text interface: a few examples of a task are provided to the model as a prompt, along with a query input, and the model generates a continuation to produce a predicted output for that query. We show that the same can be done for image and video understanding tasks such as classification, captioning, or question-answering: these can be cast as text prediction problems with visual input conditioning. The difference from a LM is that the model must be able to ingest a multimodal prompt containing images and/or videos interleaved with text. Flamingo models have this capability—they are visually-conditioned autoregressive text generation models able to ingest a sequence of text tokens interleaved with images and/or videos, and produce text as output. Flamingo models leverage two complementary pre-trained and frozen models: a vision model which can "perceive" visual scenes and a large LM which performs a basic form of reasoning. Novel architecture components are added in between these models to connect them in a way that preserves the knowledge they have accumulated during computationally intensive pre-training. Flamingo models are also able to ingest high-resolution images or videos thanks to a Perceiver-based [46] architecture that can produce a small fixed number of visual tokens per image/video, given a large and variable number of visual input features.

A crucial aspect for the performance of large LMs is that they are trained on a large amount of text data. This training provides general-purpose generation capabilities that allows these LMs to perform well when prompted with task examples. Similarly, we demonstrate that the way we train the Flamingo models is crucial for their final performance. They are trained on a carefully chosen mixture of complementary large-scale multimodal data coming only from the web, *without using any data annotated for machine learning purposes*. After this training, a Flamingo model can be directly adapted to vision tasks via simple few-shot learning without any task-specific tuning.

Contributions. In summary, our contributions are the following: **(i)** We introduce the Flamingo family of VLMs which can perform various multimodal tasks (such as captioning, visual dialogue, or visual question-answering) from only a few input/output examples. Thanks to architectural innovations, the Flamingo models can efficiently accept arbitrarily interleaved visual data and text as input and generate text in an open-ended manner. **(ii)** We quantitatively evaluate how Flamingo models can be adapted to various tasks via few-shot learning. We notably reserve a large set of held-out benchmarks which have not been used for validation of any design decisions or hyperparameters of the approach. We use these to estimate unbiased few-shot performance. **(iii)** *Flamingo* sets a new state of the art in few-shot learning on a wide array of 16 multimodal language and image/video understanding tasks. On 6 of these 16 tasks, *Flamingo* also outperforms the fine-tuned state of the art despite using only 32 task-specific examples, around 1000 times less task-specific training data than the current state of the art. With a larger annotation budget, *Flamingo* can also be effectively fine-tuned to set a new state of the art on five additional challenging benchmarks: VQAv2, VATEX, VizWiz, MSRVTTQA, and HatefulMemes.

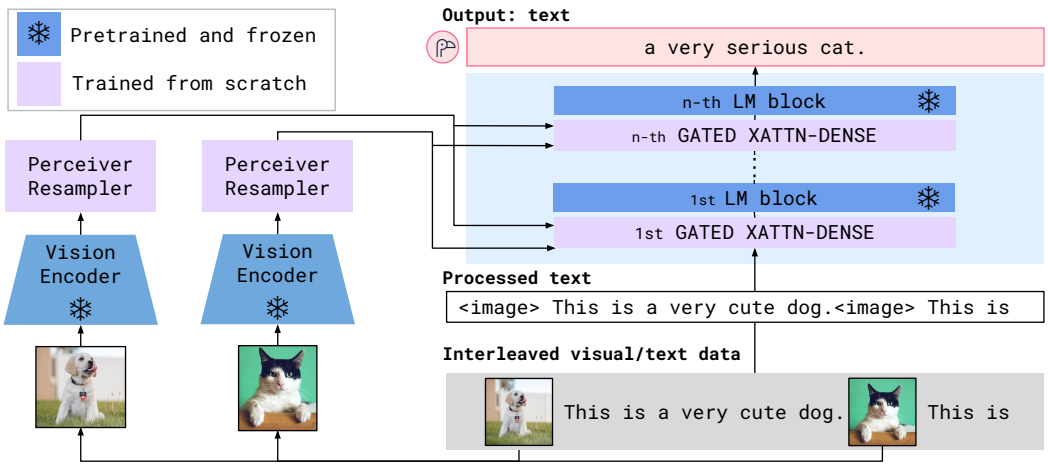

Figure 3: **Flamingo architecture overview.** Flamingo is a family of visual language models (VLMs) that take as input visual data interleaved with text and produce free-form text as output.

2 Approach

This section describes Flamingo: a visual language model that accepts text interleaved with images/videos as input and outputs free-form text. The key architectural components shown in Figure 3 are chosen to leverage pretrained vision and language models and bridge them effectively. First, the Perceiver Resampler (Section 2.1) receives spatio-temporal features from the Vision Encoder (obtained from either an image or a video) and outputs a fixed number of visual tokens. Second, these visual tokens are used to condition the frozen LM using freshly initialised cross-attention layers (Section 2.2) that are interleaved between the pretrained LM layers. These new layers offer an expressive way for the LM to incorporate visual information for the next-token prediction task. Flamingo models the likelihood of text y conditioned on interleaved images and videos x as follows:

$$p(y|x) = \prod_{\ell=1}^{L} p(y_\ell | y_{<\ell}, x_{\leq \ell}), \tag{1}$$

where y_ℓ is the ℓ-th language token of the input text, $y_{<\ell}$ is the set of preceding tokens, $x_{\leq \ell}$ is the set of images/videos preceding token y_ℓ in the interleaved sequence and p is parametrized by a Flamingo model. The ability to handle interleaved text and visual sequences (Section 2.3) makes it natural to use Flamingo models for in-context few-shot learning, analogously to GPT-3 with few-shot text prompting. The model is trained on a diverse mixture of datasets as described in Section 2.4.

2.1 Visual processing and the Perceiver Resampler

Vision Encoder: from pixels to features. Our vision encoder is a pretrained and frozen Normalizer-Free ResNet (NFNet) [10] – we use the F6 model. We pretrain the vision encoder using a contrastive objective on our datasets of image and text pairs, using the two-term contrastive loss from Radford et al. [82]. We use the output of the final stage, a 2D spatial grid of features that is flattened to a 1D sequence. For video inputs, frames are sampled at 1 FPS and encoded independently to obtain a 3D spatio-temporal grid of features to which learned temporal embeddings are added. Features are then flattened to 1D before being fed to the Perceiver Resampler. More details on the contrastive model training and performance are given in Appendix A and Appendix B, respectively.

Perceiver Resampler: from varying-size large feature maps to few visual tokens. This module connects the vision encoder to the frozen language model as shown in Figure 3. It takes as input a variable number of image or video features from the vision encoder and produces a fixed number of visual outputs (64), reducing the computational complexity of the vision-text cross-attention. Similar to Perceiver [46] and DETR [13], we learn a predefined number of latent input queries which are fed to a Transformer and cross-attend to the visual features. We show in our ablation studies (Section 3.3) that using such a vision-language resampler module outperforms a plain Transformer and an MLP. We provide an illustration, more architectural details, and pseudo-code in Appendix A.

2.2 Conditioning frozen language models on visual representations

Text generation is performed by a Transformer decoder, conditioned on the visual representations produced by the Perceiver Resampler. We interleave pretrained and frozen text-only LM blocks with blocks trained from scratch that cross-attend to the visual output from the Perceiver Resampler.

Interleaving new GATED XATTN-DENSE layers within a frozen pretrained LM. We freeze the pretrained LM blocks, and insert *gated cross-attention dense* blocks between the original layers, trained from scratch. To ensure that at initialization, the conditioned model yields the same results as the original language model, we use a $\tanh$-gating mechanism [40]. This multiplies the output of a newly added layer by $\tanh(\alpha)$ before adding it to the input representation from the residual connection, where α is a layer-specific learnable scalar initialized to 0 [4]. Thus, at initialization, the model output matches that of the pretrained LM, improving training stability and final performance. In our ablation studies (Section 3.3), we compare the proposed GATED XATTN-DENSE layers against recent alternatives [21, 65] and explore the effect of how frequently these additional layers are inserted to trade off between efficiency and expressivity. See Appendix A for more details.

Varying model sizes. We perform experiments across three models sizes, building on the 1.4B, 7B, and 70B parameter Chinchilla models [41]; calling them respectively *Flamingo*-3B, *Flamingo*-9B and *Flamingo*-80B. For brevity, we refer to the last as *Flamingo* throughout the paper. While increasing the parameter count of the frozen LM and the trainable vision-text GATED XATTN-DENSE modules, we maintain a fixed-size frozen vision encoder and trainable Perceiver Resampler across the different models (small relative to the full model size). See Appendix B for further details.

2.3 Multi-visual input support: per-image/video attention masking

The image-causal modelling introduced in Equation (1) is obtained by masking the full text-to-image cross-attention matrix, limiting which visual tokens the model sees at each text token. At a given text token, the model attends to the visual tokens of the image that appeared just before it in the interleaved sequence, rather than to all previous images (formalized and illustrated in Appendix A). Though the model only *directly* attends to a single image at a time, the dependency on all previous images remains via self-attention in the LM. This single-image cross-attention scheme importantly allows the model to seamlessly generalise to any number of visual inputs, regardless of how many are used during training. In particular, we use only up to 5 images per sequence when training on our interleaved datasets, yet our model is able to benefit from sequences of up to 32 pairs (or "shots") of images/videos and corresponding texts during evaluation. We show in Section 3.3 that this scheme is more effective than allowing the model to cross-attend to all previous images directly.

2.4 Training on a mixture of vision and language datasets

We train the Flamingo models on a mixture of three kinds of datasets, all scraped from the web: an interleaved image and text dataset derived from webpages, image-text pairs, and video-text pairs.

M3W: Interleaved image and text dataset. The few-shot capabilities of Flamingo models rely on training on interleaved text and image data. For this purpose, we collect the *MultiModal MassiveWeb* (*M3W*) dataset. We extract both text and images from the HTML of approximately 43 million webpages, determining the positions of images relative to the text based on the relative positions of the text and image elements in the Document Object Model (DOM). An example is then constructed by inserting `<image>` tags in plain text at the locations of the images on the page, and inserting a special `<EOC>` (*end of chunk*) token (added to the vocabulary and learnt) prior to any image and at the end of the document. From each document, we sample a random subsequence of $L = 256$ tokens and take up to the first $N = 5$ images included in the sampled sequence. Further images are discarded in order to save compute. More details are provided in Appendix A.

Pairs of image/video and text. We align the syntax of paired datasets with the syntax of M3W by prepending `<image>` and appending `<EOC>` to each training caption (see Appendix A for details).

Multi-objective training and optimisation strategy. We train our models by minimizing a weighted sum of per-dataset expected negative log-likelihoods of text, given the visual inputs:

$$\sum_{m=1}^{M} \lambda_m \cdot \mathbb{E}_{(x,y)\sim\mathcal{D}_m} \left[-\sum_{\ell=1}^{L} \log p(y_\ell | y_{<\ell}, x_{\leq \ell}) \right], \tag{2}$$

where $\mathcal{D}_m$ and λ_m are the m-th dataset and its weighting, respectively. Tuning the per-dataset weights λ_m is key to performance. We accumulate gradients over all datasets, which we found outperforms a "round-robin" approach [17]. We provide further training details and ablations in Appendix B.

2.5 Task adaptation with few-shot in-context learning

Once Flamingo is trained, we use it to tackle a visual task by conditioning it on a multimodal interleaved prompt. We evaluate the ability of our models to rapidly adapt to new tasks using **in-context learning**, analogously to GPT-3 [11], by interleaving support example pairs in the form of $(image, text)$ or $(video, text)$, followed by the query visual input, to build a prompt (details in Appendix A). We perform **open-ended** evaluations using beam search for decoding, and **close-ended** evaluations using our model's log-likelihood to score each possible answer. We explore **zero-shot generalization** by prompting the model with two text-only examples from the task, with no corresponding images. Evaluation hyperparameters and additional details are given in Appendix B.

3 Experiments

Our goal is to develop models that can rapidly adapt to diverse and challenging tasks. For this, we consider a wide array of 16 popular multimodal image/video and language benchmarks. In order to validate model design decisions during the course of the project, 5 of these benchmarks were used as part of our development (DEV) set: COCO, OKVQA, VQAv2, MSVDQA and VATEX. Performance estimates on the DEV benchmarks may be biased, as a result of model selection. We note that this is also the case for prior work which makes use of similar benchmarks to validate and ablate design decisions. To account for this, we report performance on an additional set of 11 benchmarks, spanning captioning, video question-answering, as well as some less commonly explored capabilities such as visual dialogue and multi-choice question-answering tasks. The evaluation benchmarks are described in Appendix B. We keep all evaluation hyperparameters fixed across all benchmarks. Depending on the task, we use four few-shot prompt templates we describe in more detail in Appendix B. We emphasize that *we do not validate any design decisions on these 11 benchmarks* and use them solely to estimate unbiased few-shot learning performance of our models.

Concretely, estimating few-shot learning performance of a model involves prompting it with a set of *support* samples and evaluating it on a set of *query* samples. For the DEV benchmarks that are used both to validate design decisions and hyperparameters, as well as to report final performance, we therefore use four subsets: *validation support*, *validation query*, *test support* and *test query*. For other benchmarks, we need only the latter two. We report in Appendix B how we form these subsets.

We report the results of the Flamingo models on few-shot learning in Section 3.1. Section 3.2 gives *Flamingo* fine-tuned results. An ablation study is given in Section 3.3. Appendix B provides more results including Flamingo's performance on the ImageNet and Kinetics700 classification tasks, and on our contrastive model's performance. Appendix C includes additional qualitative results.

3.1 Few-shot learning on vision-language tasks

Few-shot results. Results are given in Table 1. *Flamingo* outperforms by a large margin *all* previous zero-shot or few-shot methods on the 16 benchmarks considered. This is achieved with as few as four examples per task, demonstrating practical and efficient adaptation of vision models to new tasks. More importantly, *Flamingo* is often competitive with state-of-the-art methods additionally fine-tuned on up to hundreds of thousands of annotated examples. On six tasks, *Flamingo* even outperforms the fine-tuned SotA despite using a *single* set of model weights and only 32 task-specific examples. Finally, despite having only used the DEV benchmarks for design decisions, our results generalize well to the other benchmarks, confirming the generality of our approach.

Scaling with respect to parameters and shots. As shown in Figure 2, the larger the model, the better the few-shot performance, similar to GPT-3 [11]. The performance also improves with the number of shots. We further find that the largest model better exploits larger numbers of shots. Interestingly, even though our Flamingo models were trained with sequences limited to only 5 images on *M3W*, they are still able to benefit from up to 32 images or videos during inference. This demonstrates the flexibility of the Flamingo architecture for processing a variable number of videos or images.

Method	FT	Shot	OKVQA (I)	VQAv2 (I)	COCO (I)	MSVDQA (V)	VATEX (V)	VizWiz (I)	Flick30K (I)	MSRVTTQA (V)	iVQA (V)	YouCook2 (V)	STAR (V)	VisDial (I)	TextVQA (I)	NextQA (I)	HatefulMemes (I)	RareAct (V)
Zero/Few shot SOTA			[33]	[110]	[120]	[55]				[55]	[131]		[139]	[76]			[82]	[82]
	✗		43.3	38.2	32.2	35.2	-	-	-	19.2	12.2	-	39.4	11.6	-	-	66.1	40.7
		(X)	(16)	(4)	(0)	(0)				(0)	(0)		(0)	(0)			(0)	(0)
Flamingo-3B	✗	0	41.2	49.2	73.0	27.5	40.1	28.9	60.6	11.0	32.7	55.8	39.6	46.1	30.1	21.3	53.7	58.4
	✗	4	43.3	53.2	85.0	33.0	50.0	34.0	72.0	14.9	35.7	64.6	41.3	47.3	32.7	22.4	53.6	-
	✗	32	45.9	57.1	99.0	42.6	59.2	45.5	71.2	25.6	37.7	76.7	41.6	47.3	30.6	26.1	56.3	-
Flamingo-9B	✗	0	44.7	51.8	79.4	30.2	39.5	28.8	61.5	13.7	35.2	55.0	41.8	48.0	31.8	23.0	57.0	57.9
	✗	4	49.3	56.3	93.1	36.2	51.7	34.9	72.6	18.2	37.7	70.8	**42.8**	50.4	33.6	24.7	62.7	-
	✗	32	51.0	60.4	106.3	47.2	57.4	44.0	72.8	29.4	40.7	77.3	41.2	50.4	32.6	28.4	63.5	-
Flamingo	✗	0	50.6	56.3	84.3	35.6	46.7	31.6	67.2	17.4	40.7	60.1	39.7	52.0	35.0	26.7	46.4	**60.8**
	✗	4	57.4	63.1	103.2	41.7	56.0	39.6	75.1	23.9	44.1	74.5	42.4	**55.6**	36.5	30.8	68.6	-
	✗	32	**57.8**	**67.6**	**113.8**	**52.3**	**65.1**	**49.8**	**75.4**	**31.0**	**45.3**	**86.8**	42.2	**55.6**	**37.9**	**33.5**	**70.0**	-
Pretrained FT SOTA	✔		54.4	80.2	143.3	47.9	76.3	57.2	67.4	46.8	35.4	138.7	36.7	75.2	54.7	25.2	79.1	
			[33]	[136]	[120]	[27]	[149]	[62]	[146]	[49]	[131]	[128]	[124]	[76]	[133]	[125]	[59]	-
		(X)	(10K)	(444K)	(500K)	(27K)	(500K)	(20K)	(30K)	(130K)	(6K)	(10K)	(46K)	(123K)	(20K)	(38K)	(9K)	

Table 1: **Comparison to the state of the art.** A *single* Flamingo model reaches the state of the art on a wide array of image **(I)** and video **(V)** understanding tasks with few-shot learning, significantly outperforming previous best zero- and few-shot methods with as few as four examples. More importantly, using only 32 examples and without adapting any model weights, Flamingo *outperforms* the current best methods – fine-tuned on thousands of annotated examples – on seven tasks. Best few-shot numbers are in **bold**, best numbers overall are underlined.

Method	VQAV2		COCO	VATEX	VizWiz		MSRVTTQA	VisDial		YouCook2	TextVQA		HatefulMemes
	test-dev	test-std	test	test	test-dev	test-std	test	valid	test-std	valid	valid	test-std	test seen
32 shots	67.6	-	113.8	65.1	49.8	-	31.0	56.8	-	86.8	36.0	-	70.0
Fine-tuned	**82.0**	**82.1**	138.1	**84.2**	**65.7**	**65.4**	**47.4**	61.8	59.7	118.6	**57.1**	54.1	**86.6**
SotA	81.3†	81.3†	**149.6**†	81.4†	57.2†	60.6†	46.8	**75.2**	**75.4**†	**138.7**	54.7	**73.7**	84.6†
	[129]	[129]	[115]	[149]	[62]	[62]	[49]	[76]	[119]	[128]	[133]	[81]	[148]

Table 2: **Comparison to SotA when fine-tuning *Flamingo*.** We fine-tune *Flamingo* on all nine tasks where *Flamingo* does not achieve SotA with few-shot learning. *Flamingo* sets a new SotA on five of them, outperfoming methods (marked with †) that use tricks such as model ensembling or domain-specific metric optimisation (e.g., CIDEr optimisation).

3.2 Fine-tuning *Flamingo* as a pretrained vision-language model

While not the main focus of our work, we verify that when given more data, Flamingo models can be adapted to a task by fine-tuning their weights. In Table 2, we explore fine-tuning our largest model, *Flamingo*, for a given task with no limit on the annotation budget. In short, we do so by fine-tuning the model on a short schedule with a small learning rate by additionally unfreezing the vision backbone to accommodate a higher input resolution (details in Appendix B). We find that we can improve results over our previously presented in-context few-shot learning results, setting a new state of the art on five additional tasks: VQAv2, VATEX, VizWiz, MSRVTTQA, and HatefulMemes.

3.3 Ablation studies

In Table 3, we report our ablation results using *Flamingo*-3B on the *validation* subsets of the five DEV benchmarks with 4 shots. Note that we use smaller batch sizes and a shorter training schedule compared to the final models. The **Overall score** is obtained by dividing each benchmark score by its state-of-the-art (SotA) performance from Table 1 and averaging the results. More details and results are given in Appendix B.

Importance of the training data mixture. As shown in row **(i)**, getting the right training data plays a crucial role. In fact, removing the interleaved image-text dataset *M3W* leads to a *decrease of more than* 17% in performance while removing the conventional paired image-text pairs also decreases performance (by 9.8%), demonstrating the need for different types of datasets. Moreover, removing our paired video-text dataset negatively affects performance on all video tasks. We show in row **(ii)** the importance of our gradient accumulation strategy compared to using round-robin updates [17].

Visual conditioning of the frozen LM. We ablate the use of the 0-initialized tanh gating when merging the cross-attention output to the frozen LM output in row **(iii)**. Without it, we see a drop of 4.2% in our overall score. Moreover, we have noticed that disabling the 0-initialized tanh gating leads to training instabilities. Next, we ablate different conditioning architectures in row **(iv)**. VANILLA

	Ablated setting	*Flamingo*-3B original value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	MSVDQA top1↑	VATEX CIDEr↑	Overall score↑
		***Flamingo*-3B model**		3.2B	1.74s	86.5	42.1	55.8	36.3	53.4	**70.7**
(i)	Training data	All data	w/o Video-Text pairs	3.2B	1.42s	84.2	43.0	53.9	34.5	46.0	67.3
			w/o Image-Text pairs	3.2B	0.95s	66.3	39.2	51.6	32.0	41.6	60.9
			w/o M3W	3.2B	1.02s	54.1	36.5	52.7	31.4	23.5	53.4
(ii)	Optimisation	Accumulation	Round Robin	3.2B	1.68s	76.1	39.8	52.1	33.2	40.8	62.9
(iii)	Tanh gating	✓	✗	3.2B	1.74s	78.4	40.5	52.9	35.9	47.5	66.5
(iv)	Cross-attention architecture	GATED XATTN-DENSE	VANILLA XATTN	2.4B	1.16s	80.6	41.5	53.4	32.9	50.7	66.9
			GRAFTING	3.3B	1.74s	79.2	36.1	50.8	32.2	47.8	63.1
(v)	Cross-attention frequency	Every	Single in middle	2.0B	0.87s	71.5	38.1	50.2	29.1	42.3	59.8
			Every 4th	2.3B	1.02s	82.3	42.7	55.1	34.6	50.8	68.8
			Every 2nd	2.6B	1.24s	83.7	41.0	55.8	34.5	49.7	68.2
(vi)	Resampler	Perceiver	MLP	3.2B	1.85s	78.6	42.2	54.7	35.2	44.7	66.6
			Transformer	3.2B	1.81s	83.2	41.7	55.6	31.5	48.3	66.7
(vii)	Vision encoder	NFNet-F6	CLIP ViT-L/14	3.1B	1.58s	76.5	41.6	53.4	33.2	44.5	64.9
			NFNet-F0	2.9B	1.45s	73.8	40.5	52.8	31.1	42.9	62.7
(viii)	Freezing LM	✓	✗ (random init)	3.2B	2.42s	74.8	31.5	45.6	26.9	50.1	57.8
			✗ (pretrained)	3.2B	2.42s	81.2	33.7	47.4	31.0	53.9	62.7

Table 3: **Ablation studies.** Each row should be compared to the baseline Flamingo run (top row). Step time measures the time spent to perform gradient updates on all training datasets.

XATTN, refers to the vanilla cross-attention from the original Transformer decoder [111]. In the GRAFTING approach from [65], the frozen LM is used as is with no additional layers inserted, and a stack of interleaved self-attention and cross-attention layers that take the frozen LM output are learnt from scratch. Overall, we show that our GATED XATTN-DENSE conditioning approach works best.

Compute/Memory vs. performance trade-offs. In row **(v)**, we ablate the frequency at which we add new GATED XATTN-DENSE blocks. Although adding them at every layer is better, it significantly increases the number of trainable parameters and time complexity of the model. Notably, inserting them every fourth block accelerates training by 66% while only decreasing the overall score by 1.9%. In light of this trade-off, we maximize the number of added layers under hardware constraints and add a GATED XATTN-DENSE every fourth layer for *Flamingo*-9B and every seventh for *Flamingo*-80B. We further compare in row **(vi)** the Perceiver Resampler to a MLP and a vanilla Transformer given a parameter budget. Both underperform the Perceiver Resampler while also being slower.

Vision encoder. In row **(vii)**, we compare our NFNet-F6 vision encoder pretrained with contrastive learning (details in Appendix A) to the publicly available CLIP ViT-L/14 [82] model trained at 224 resolution. Our NFNet-F6 has a $+5.8\%$ advantage over the CLIP ViT-L/14 and $+8.0\%$ over a smaller NFNet-F0 encoder, which highlights the importance of using a strong vision backbone.

Freezing LM components prevents catastrophic forgetting. We verify the importance of freezing the LM layers at training in row **(viii)**. If trained from scratch, we observe a large performance decrease of -12.9%. Interestingly, fine-tuning our pretrained LM also leads to a drop in performance of -8.0%. This indicates an instance of "catastrophic forgetting" [68], in which the model progressively forgets its pretraining while training on a new objective. In our setting, freezing the language model is a better alternative to training with the pre-training dataset (MassiveText) in the mixture.

4 Related work

Language modelling and few-shot adaptation. Language modelling has recently made substantial progress following the introduction of Transformers [111]. The paradigm of first pretraining on a vast amount of data followed by an adaptation on a downstream task has become standard [11, 22, 31, 43, 50, 72, 84, 104]. In this work, we build on the 70B Chinchilla language model [41] as the base LM for *Flamingo*. Numerous works have explored techniques to adapt language models to novel tasks using a few examples. These include adding small adapter modules [42], fine-tuning a small part of the LM [137], showing in-context examples in the prompt [11], or optimizing the prompt [53, 57] through gradient descent. In this paper, we take inspiration from the in-context [11] few-shot learning technique instead of more involved few-shot learning approaches based on metric learning [23, 99, 108, 113] or meta-learning [6, 7, 26, 30, 88, 151].

When language meets vision. These LM breakthroughs have been influential for vision-language modelling. In particular, BERT [22] inspired a large body of vision-language work [16, 27, 28, 37, 56, 58, 63, 97, 102, 103, 105, 114, 117, 138, 139, 147]. We differ from these approaches as Flamingo

models do not require fine-tuning on new tasks. Another family of vision-language models is based on contrastive learning [2, 5, 47, 48, 54, 71, 79, 82, 134, 136, 142]. Flamingo differs from contrastive models as it can generate text, although we build and rely upon them for our vision encoder. Similar to our work are VLMs able to generate text in an autoregressive manner [24, 44, 64, 112]. Concurrent works [17, 55, 115, 120, 150] also propose to formulate numerous vision tasks as text generation problems. Building on top of powerful pretrained language models has been explored in several recent works. One recent line of work [25, 65, 75, 110, 132, 140] proposes to freeze the pretrained LM weights to prevent catastrophic forgetting [68]. We follow this idea by freezing the Chinchilla LM layers [41] and adding learnable layers within the frozen LM. We differ from prior work by introducing the first LM that can ingest arbitrarily interleaved images, videos, and text.

Web-scale vision and language training datasets. Manually annotated vision and language datasets are costly to obtain and thus relatively small (10k-100k) in scale [3, 15, 66, 118, 125, 135]. To alleviate this lack of data, numerous works [14, 48, 94, 106] automatically scrape readily available paired vision-text data. In addition to such paired data, we show the importance of also training on entire multimodal webpages containing interleaved images and text as a single sequence. CM3 [1] follows a similar approach, additionally generating the full HTML markup from pages, while we simplify the text prediction task for the base LM by scraping only the natural language text.

5 Discussion

Limitations. First, our models build on pretrained LMs, and as a side effect, directly inherit their weaknesses. For example, LM priors are generally helpful, but may play a role in occasional hallucinations and ungrounded guesses. Furthermore, LMs generalise poorly to sequences longer than the training ones. They also suffer from poor sample efficiency during training. Addressing these issues can accelerate progress in the field and enhance the abilities of VLMs like Flamingo.

Second, the classification performance of Flamingo lags behind that of state-of-the-art contrastive models [79, 82]. These models directly optimize for text-image retrieval, of which classification is a special case. In contrast, our models handle a wider range of tasks, such as open-ended ones. A unified approach to achieve the best of both worlds is an important research direction.

Third, in-context learning has significant advantages over gradient-based few-shot learning methods, but also suffers from drawbacks depending on the characteristics of the application at hand. We demonstrate the effectiveness of in-context learning when access is limited to only a few dozen examples. In-context learning also enables simple deployment, requiring only inference, generally with no hyperparameter tuning needed. However, in-context learning is known to be highly sensitive to various aspects of the demonstrations [77, 144], and its inference compute cost and absolute performance scale poorly with the number of shots beyond this low-data regime. There may be opportunities to combine few-shot learning methods to leverage their complementary benefits. We discuss the limitations of our work in more depth in Appendix D.

Societal impacts. In terms of societal impacts, *Flamingo* offers a number of benefits while carrying some risks. Its ability to rapidly adapt to a broad range of tasks have the potential to enable non-expert users to obtain good performance in data-starved regimes, lowering the barriers to both beneficial and malicious applications. *Flamingo* is exposed to the same risks as large language models, such as outputting offensive language, propagating social biases and stereotypes, as well as leaking private information [41, 122]. Its ability to additionally handle visual inputs poses specific risks such as gender and racial biases relating to the contents of the input images, similar to a number of visual recognition systems [12, 20, 36, 93, 143]. We refer the reader to Appendix D for a more extensive discussion of the societal impacts of our work, both positive and negative; as well as mitigation strategies and early investigations of risks relating to racial or gender bias and toxic outputs. Finally we note that, following prior work focusing on language models [69, 78, 107], the few-shot capabilities of Flamingo could be useful for mitigating such risks.

Conclusion. We proposed Flamingo, a general-purpose family of models that can be applied to image and video tasks with minimal task-specific training data. We also qualitatively explored interactive abilities of *Flamingo* such as "chatting" with the model, demonstrating flexibility beyond traditional vision benchmarks. Our results suggest that connecting pre-trained large language models with powerful visual models is an important step towards general-purpose visual understanding.

References

[1] Armen Aghajanyan, Bernie Huang, Candace Ross, Vladimir Karpukhin, Hu Xu, Naman Goyal, Dmytro Okhonko, Mandar Joshi, Gargi Ghosh, Mike Lewis, and Luke Zettlemoyer. CM3: A causal masked multimodal model of the internet. *arXiv:2201.07520*, 2022.

[2] Jean-Baptiste Alayrac, Adria Recasens, Rosalia Schneider, Relja Arandjelović, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. *Conference on Neural Information Processing Systems*, 2020.

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *International Conference on Computer Vision*, 2015.

[4] Thomas Bachlechner, Bodhisattwa Prasad Majumder, Henry Mao, Gary Cottrell, and Julian McAuley. ReZero is all you need: Fast convergence at large depth. In *Uncertainty in Artificial Intelligence*, 2021.

[5] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *International Conference on Computer Vision*, 2021.

[6] Luca Bertinetto, João F. Henriques, Jack Valmadre, Philip Torr, and Andrea Vedaldi. Learning feed-forward one-shot learners. *Conference on Neural Information Processing Systems*, 2016.

[7] Luca Bertinetto, Joao F. Henriques, Philip H. S. Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *arXiv:1805.08136*, 2018.

[8] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL http://github.com/google/jax.

[9] John S. Bridle. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In *Neurocomputing*, 1990.

[10] Andrew Brock, Soham De, Samuel L. Smith, and Karen Simonyan. High-performance large-scale image recognition without normalization. *arXiv:2102.06171*, 2021.

[11] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Conference on Neural Information Processing Systems*, 2020.

[12] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *ACM Conference on Fairness, Accountability, and Transparency*, 2018.

[13] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, 2020.

[14] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *IEEE Computer Vision and Pattern Recognition*, 2021.

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

[16] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. UNITER: Universal image-text representation learning. In *European Conference on Computer Vision*, 2020.

[17] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *International Conference on Machine Learning*, 2021.

[18] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. PaLM: Scaling language modeling with pathways. *arXiv:2204.02311*, 2022.

[19] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *IEEE Computer Vision and Pattern Recognition*, 2017.

[20] Terrance De Vries, Ishan Misra, Changhan Wang, and Laurens Van der Maaten. Does object recognition work for everyone? In *IEEE Computer Vision and Pattern Recognition*, 2019.

[21] Karan Desai and Justin Johnson. VirTex: Learning visual representations from textual annotations. In *IEEE Computer Vision and Pattern Recognition*, 2021.

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

[23] Carl Doersch, Ankush Gupta, and Andrew Zisserman. CrossTransformers: spatially-aware few-shot transfer. *Conference on Neural Information Processing Systems*, 2020.

[24] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *IEEE Computer Vision and Pattern Recognition*, 2015.

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

[26] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 2017.

[27] Tsu-Jui Fu, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. VIOLET: End-to-end video-language transformers with masked visual-token modeling. *arXiv:2111.12681*, 2021.

[28] Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-scale adversarial training for vision-and-language representation learning. In *Conference on Neural Information Processing Systems*, 2020.

[29] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 2021.

[30] Jonathan Gordon, John Bronskill, Matthias Bauer, Sebastian Nowozin, and Richard E. Turner. Meta-learning probabilistic inference for prediction. *arXiv:1805.09921*, 2018.

[31] Alex Graves. Generating sequences with recurrent neural networks. *arXiv:1308.0850*, 2013.

[32] Thomas L. Griffiths, Frederick Callaway, Michael B. Chang, Erin Grant, Paul M. Krueger, and Falk Lieder. Doing more with less: meta-reasoning and meta-learning in humans and machines. *Current Opinion in Behavioral Sciences*, 2019.

[33] Liangke Gui, Borui Wang, Qiuyuan Huang, Alex Hauptmann, Yonatan Bisk, and Jianfeng Gao. KAT: A knowledge augmented transformer for vision-and-language. *arXiv:2112.08614*, 2021.

[34] Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *IEEE Computer Vision and Pattern Recognition*, 2018.

[35] Adi Haviv, Ori Ram, Ofir Press, Peter Izsak, and Omer Levy. Transformer language models without positional encodings still learn positional information. *arXiv:2203.16634*, 2022.

[36] Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *European Conference on Computer Vision*, 2018.

[37] Lisa Anne Hendricks, John Mellor, Rosalia Schneider, Jean-Baptiste Alayrac, and Aida Nematzadeh. Decoupling the role of data, attention, and losses in multimodal transformers. *Annual Meeting of the Association for Computational Linguistics*, 2021.

[38] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv:1606.08415*, 2016.

[39] Tom Hennigan, Trevor Cai, Tamara Norman, and Igor Babuschkin. Haiku: Sonnet for JAX, 2020. URL http://github.com/deepmind/dm-haiku.

[40] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 1997.

[41] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Eric Noland Tom Hennigan, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. *arXiv:2203.15556*, 2022.

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

[43] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. *arXiv:1801.06146*, 2018.

[44] Xiaowei Hu, Zhe Gan, Jianfeng Wang, Zhengyuan Yang, Zicheng Liu, Yumao Lu, and Lijuan Wang. Scaling up vision-language pre-training for image captioning. *arXiv:2111.12233*, 2021.

[45] Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. Attention on attention for image captioning. In *International Conference on Computer Vision*, 2019.

[46] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International Conference on Machine Learning*, 2021.

[47] Aashi Jain, Mandy Guo, Krishna Srinivasan, Ting Chen, Sneha Kudugunta, Chao Jia, Yinfei Yang, and Jason Baldridge. MURAL: multimodal, multitask retrieval across languages. *arXiv:2109.05125*, 2021.

[48] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. *arXiv:2102.05918*, 2021.

[49] Alex Jinpeng Wang, Yixiao Ge, Rui Yan, Yuying Ge, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. All in one: Exploring unified video-language pre-training. *arXiv:2203.07303*, 2022.

[50] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling. *arXiv:1602.02410*, 2016.

[51] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv:2001.08361*, 2020.

[52] Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The Hateful Memes Challenge: Detecting hate speech in multimodal memes. *Conference on Neural Information Processing Systems*, 2020.

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

[54] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *Conference on Neural Information Processing Systems*, 2021.

[55] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv:2201.12086*, 2022.

[56] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. HERO: Hierarchical encoder for video+language omni-representation pre-training. *arXiv:2005.00200*, 2020.

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

[58] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, 2020.

[59] Phillip Lippe, Nithin Holla, Shantanu Chandra, Santhosh Rajamanickam, Georgios Antoniou, Ekaterina Shutova, and Helen Yannakoudakis. A multimodal framework for the detection of hateful memes. *arXiv:2012.12871*, 2020.

[60] Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. What makes good in-context examples for GPT-3? *arXiv:2101.06804*, 2021.

[61] Siqi Liu, Zhenhai Zhu, Ning Ye, Sergio Guadarrama, and Kevin Murphy. Optimization of image description metrics using policy gradient methods. In *International Conference on Computer Vision*, 2017.

[62] Yu Liu, Lianghua Huang, Liuyihang Song, Bin Wang, Yingya Zhang, and Pan Pan. Enhancing textual cues in multi-modal transformers for VQA. *VizWiz Challenge 2021*, 2021.

[63] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Conference on Neural Information Processing Systems*, 2019.

[64] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. UniVL: A unified video and language pre-training model for multimodal understanding and generation. *arXiv:2002.06353*, 2020.

[65] Ziyang Luo, Yadong Xi, Rongsheng Zhang, and Jing Ma. VC-GPT: Visual conditioned GPT for end-to-end generative vision-and-language pre-training. *arXiv:2201.12723*, 2022.

[66] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *IEEE Computer Vision and Pattern Recognition*, 2019.

[67] Ellen M. Markman. *Categorization and naming in children: Problems of induction*. MIT Press, 1989.

[68] Michael McCloskey and Neil J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *The Psychology of Learning and Motivation*, 1989.

[69] Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, and Nat McAleese. Teaching language models to support answers with verified quotes. *arXiv:2203.11147*, 2022.

[70] Antoine Miech, Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Andrew Zisserman. RareAct: A video dataset of unusual interactions. *arxiv:2008.01018*, 2020.

[71] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *IEEE Computer Vision and Pattern Recognition*, 2020.

[72] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. *Interspeech*, 2010.

[73] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv:2202.12837*, 2022.

[74] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *ACM Conference on Fairness, Accountability, and Transparency*, 2019.

[75] Ron Mokady, Amir Hertz, and Amit H. Bermano. ClipCap: CLIP prefix for image captioning. *arXiv:2111.09734*, 2021.

[76] Vishvak Murahari, Dhruv Batra, Devi Parikh, and Abhishek Das. Large-scale pretraining for visual dialog: A simple state-of-the-art baseline. In *European Conference on Computer Vision*, 2020.

[77] Ethan Perez, Douwe Kiela, and Kyunghyun Cho. True few-shot learning with language models. *Conference on Neural Information Processing Systems*, 2021.

[78] Ethan Perez, Saffron Huang, Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. *arXiv:2202.03286*, 2022.

[79] Hieu Pham, Zihang Dai, Golnaz Ghiasi, Hanxiao Liu, Adams Wei Yu, Minh-Thang Luong, Mingxing Tan, and Quoc V. Le. Combined scaling for zero-shot transfer learning. *arXiv:2111.10050*, 2021.

[80] Ofir Press, Noah Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. In *International Conference on Learning Representations*, 2022.

[81] Yixuan Qiao, Hao Chen, Jun Wang, Yihao Chen, Xianbin Ye, Ziliang Li, Xianbiao Qi, Peng Gao, and Guotong Xie. Winner team Mia at TextVQA Challenge 2021: Vision-and-language representation learning with pre-trained sequence-to-sequence model. *arXiv:2106.15332*, 2021.

[82] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *arXiv:2103.00020*, 2021.

[83] Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. Scaling language models: Methods, analysis & insights from training Gopher. *arXiv:2112.11446*, 2021.

[84] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv:1910.10683*, 2019.

[85] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. ZeRO: Memory optimizations toward training trillion parameter models. In *International Conference for High Performance Computing, Networking, Storage and Analysis*, 2020.

[86] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv:2204.06125*, 2022.

[87] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *IEEE Computer Vision and Pattern Recognition*, 2017.

[88] James Requeima, Jonathan Gordon, John Bronskill, Sebastian Nowozin, and Richard E. Turner. Fast and flexible multi-task classification using conditional neural adaptive processes. *Conference on Neural Information Processing Systems*, 2019.

[89] Laria Reynolds and Kyle McDonell. 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*, 2021.

[90] Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. Gender bias in coreference resolution. *arXiv:1804.09301*, 2018.

[91] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015.

[92] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M. Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. Multitask Prompted Training Enables Zero-Shot Task Generalization. In *International Conference on Learning Representations*, 2022.

[93] Carsten Schwemmer, Carly Knight, Emily D. Bello-Pardo, Stan Oklobdzija, Martijn Schoonvelde, and Jeffrey W. Lockhart. Diagnosing gender bias in image recognition systems. *Socius*, 2020.

[94] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual Captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Annual Meeting of the Association for Computational Linguistics*, 2018.

[95] Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. Megatron-LM: Training multi-billion parameter language models using model parallelism. *arXiv:2104.08691*, 2019.

[96] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *IEEE Computer Vision and Pattern Recognition*, 2019.

[97] Amanpreet Singh, Ronghang Hu, Vedanuj Goswami, Guillaume Couairon, Wojciech Galuba, Marcus Rohrbach, and Douwe Kiela. FLAVA: A foundational language and vision alignment model. *arXiv:2112.04482*, 2021.

[98] Lucas Smaira, João Carreira, Eric Noland, Ellen Clancy, Amy Wu, and Andrew Zisserman. A short note on the Kinetics-700-2020 human action dataset. *arXiv:2010.10864*, 2020.

[99] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. *Conference on Neural Information Processing Systems*, 2017.

[100] David R So, Wojciech Mańke, Hanxiao Liu, Zihang Dai, Noam Shazeer, and Quoc V. Le. Primer: Searching for efficient transformers for language modeling. *arXiv:2109.08668*, 2021.

[101] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and policy considerations for deep learning in NLP. *arXiv:1906.02243*, 2019.

[102] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. VL-BERT: Pre-training of generic visual-linguistic representations. *arXiv:1908.08530*, 2019.

[103] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. VideoBERT: A joint model for video and language representation learning. In *International Conference on Computer Vision*, 2019.

[104] Ilya Sutskever, James Martens, and Geoffrey E. Hinton. Generating text with recurrent neural networks. In *International Conference on Machine Learning*, 2011.

[105] Hao Tan and Mohit Bansal. LXMERT: Learning cross-modality encoder representations from transformer. In *Conference on Empirical Methods in Natural Language Processing*, 2019.

[106] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. YFCC100M: The new data in multimedia research. *Communications of the ACM*, 2016.

[107] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, YaGuang Li, Hongrae Lee, Huaixiu Steven Zheng, Amin Ghafouri, Marcelo Menegali, Yanping Huang, Maxim Krikun, Dmitry Lepikhin, James Qin, Dehao Chen, Yuanzhong Xu, Zhifeng Chen, Adam Roberts, Maarten Bosma, Vincent Zhao, Yanqi Zhou, Chung-Ching Chang, Igor Krivokon, Will Rusch, Marc Pickett, Pranesh Srinivasan, Laichee Man, Kathleen Meier-Hellstern, Meredith Ringel Morris, Tulsee Doshi, Renelito Delos Santos, Toju Duke, Johnny Soraker, Ben Zevenbergen, Vinodkumar Prabhakaran, Mark Diaz, Ben Hutchinson, Kristen Olson, Alejandra Molina, Erin Hoffman-John, Josh Lee, Lora Aroyo, Ravi Rajakumar, Alena Butryna, Matthew Lamm, Viktoriya Kuzmina, Joe Fenton, Aaron Cohen, Rachel Bernstein, Ray Kurzweil, Blaise Aguera-Arcas, Claire Cui, Marian Croak, Ed Chi, and Quoc Le. LaMDA: Language models for dialog applications. *arXiv:2201.08239*, 2022.

[108] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B. Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? In *European Conference on Computer Vision*, 2020.

[109] Hugo Touvron, Andrea Vedaldi, Matthijs Douze, and Hervé Jégou. Fixing the train-test resolution discrepancy. *Conference on Neural Information Processing Systems*, 2019.

[110] Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. *Conference on Neural Information Processing Systems*, 2021.

[111] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Conference on Neural Information Processing Systems*, 2017.

[112] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *International Conference on Computer Vision*, 2015.

[113] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *Conference on Neural Information Processing Systems*, 2016.

[114] Jianfeng Wang, Xiaowei Hu, Zhe Gan, Zhengyuan Yang, Xiyang Dai, Zicheng Liu, Yumao Lu, and Lijuan Wang. UFO: A unified transformer for vision-language representation learning. *arXiv:2111.10023*, 2021.

[115] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. *arXiv:2202.03052*, 2022.

[116] Thomas Wang, Adam Roberts, Daniel Hesslow, Teven Le Scao, Hyung Won Chung, Iz Beltagy, Julien Launay, and Colin Raffel. What language model architecture and pretraining objective work best for zero-shot generalization? *arXiv:2204.05832*, 2022.

[117] Wenhui Wang, Hangbo Bao, Li Dong, and Furu Wei. VLMo: Unified vision-language pre-training with mixture-of-modality-experts. *arXiv:2111.02358*, 2021.

[118] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. VATEX: A large-scale, high-quality multilingual dataset for video-and-language research. In *International Conference on Computer Vision*, 2019.

[119] Yue Wang, Shafiq Joty, Michael Lyu, Irwin King, Caiming Xiong, and Steven Hoi. VD-BERT: A unified vision and dialog transformer with BERT. In *Conference on Empirical Methods in Natural Language Processing*, 2020.

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

[121] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. *arXiv:2109.01652*, 2021.

[122] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, Zac Kenton, Sasha Brown, Will Hawkins, Tom Stepleton, Courtney Biles, Abeba Birhane, Julia Haas, Laura Rimell, Lisa Anne Hendricks, William Isaac, Sean Legassick, Geoffrey Irving, and Iason Gabriel. Ethical and social risks of harm from language models. *arXiv:2112.04359*, 2021.

[123] Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. *arXiv:2203.05482*, 2022.

[124] Bo Wu, Shoubin Yu, Zhenfang Chen, Joshua B. Tenenbaum, and Chuang Gan. STAR: A Benchmark for Situated Reasoning in Real-World Videos. In *Conference on Neural Information Processing Systems*, 2021.

[125] Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-QA: Next phase of question-answering to explaining temporal actions. In *IEEE Computer Vision and Pattern Recognition*, 2021.

[126] Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *ACM Multimedia*, 2017.

[127] Hanwei Xu, Yujun Chen, Yulun Du, Nan Shao, Yanggang Wang, Haiyu Li, and Zhilin Yang. Zeroprompt: Scaling prompt-based pretraining to 1,000 tasks improves zero-shot generalization. *arXiv:2201.06910*, 2022.

[128] Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. VLM: Task-agnostic video-language model pre-training for video understanding. *arXiv:2105.09996*, 2021.

[129] Ming Yan, Haiyang Xu, Chenliang Li, Junfeng Tian, Bin Bi, Wei Wang, Weihua Chen, Xianzhe Xu, Fan Wang, Zheng Cao, Zhicheng Zhang, Qiyu Zhang, Ji Zhang, Songfang Huang, Fei Huang, Luo Si, and Rong Jin. Achieving human parity on visual question answering. *arXiv:2111.08896*, 2021.

[130] Shen Yan, Xuehan Xiong, Anurag Arnab, Zhichao Lu, Mi Zhang, Chen Sun, and Cordelia Schmid. Multiview transformers for video recognition. *arXiv:2201.04288*, 2022.

[131] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Just ask: Learning to answer questions from millions of narrated videos. In *International Conference on Computer Vision*, 2021.

[132] Zhengyuan Yang, Zhe Gan, Jianfeng Wang, Xiaowei Hu, Yumao Lu, Zicheng Liu, and Lijuan Wang. An empirical study of GPT-3 for few-shot knowledge-based VQA. In *National Conference on Artificial Intelligence (AAAI)*, 2021.

[133] Zhengyuan Yang, Yijuan Lu, Jianfeng Wang, Xi Yin, Dinei Florencio, Lijuan Wang, Cha Zhang, Lei Zhang, and Jiebo Luo. TAP: Text-aware pre-training for text-VQA and text-caption. In *IEEE Computer Vision and Pattern Recognition*, 2021.

[134] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. FILIP: Fine-grained interactive language-image pre-training. *arXiv:2111.07783*, 2021.

[135] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Annual Meeting of the Association for Computational Linguistics*, 2014.

[136] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, Ce Liu, Mengchen Liu, Zicheng Liu, Yumao Lu, Yu Shi, Lijuan Wang, Jianfeng Wang, Bin Xiao, Zhen Xiao, Jianwei Yang, Michael Zeng, Luowei Zhou, and Pengchuan Zhang. Florence: A new foundation model for computer vision. *arXiv:2111.11432*, 2021.

[137] Elad Ben Zaken, Shauli Ravfogel, and Yoav Goldberg. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. *arXiv:2106.10199*, 2021.

[138] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. MERLOT: Multimodal neural script knowledge models. *Conference on Neural Information Processing Systems*, 2021.

[139] Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. MERLOT reserve: Neural script knowledge through vision and language and sound. In *IEEE Computer Vision and Pattern Recognition*, 2022.

[140] Andy Zeng, Adrian Wong, Stefan Welker, Krzysztof Choromanski, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, and Pete Florence. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv:2204.00598*, 2022.

[141] Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers. *arXiv:2106.04560*, 2021.

[142] Xiaohua Zhai, Xiao Wang, Basil Mustafa, Andreas Steiner, Daniel Keysers, Alexander Kolesnikov, and Lucas Beyer. LiT: Zero-shot transfer with locked-image text tuning. *arXiv:2111.07991*, 2021.

[143] Dora Zhao, Angelina Wang, and Olga Russakovsky. Understanding and evaluating racial biases in image captioning. In *IEEE Computer Vision and Pattern Recognition*, 2021.

[144] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, 2021.

[145] Luowei Zhou, Chenliang Xu, and Jason J. Corso. Towards automatic learning of procedures from web instructional videos. In *National Conference on Artificial Intelligence (AAAI)*, 2018.

[146] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and VQA. In *National Conference on Artificial Intelligence (AAAI)*, 2020.

[147] Linchao Zhu and Yi Yang. ActBERT: Learning global-local video-text representations. In *IEEE Computer Vision and Pattern Recognition*, 2020.

[148] Ron Zhu. Enhance multimodal transformer with external label and in-domain pretrain: Hateful meme challenge winning solution. *arXiv:2012.08290*, 2020.

[149] Xinxin Zhu, Longteng Guo, Peng Yao, Shichen Lu, Wei Liu, and Jing Liu. Vatex video captioning challenge 2020: Multi-view features and hybrid reward strategies for video captioning. *arXiv:1910.11102*, 2019.

[150] Xizhou Zhu, Jinguo Zhu, Hao Li, Xiaoshi Wu, Xiaogang Wang, Hongsheng Li, Xiaohua Wang, and Jifeng Dai. Uni-Perceiver: Pre-training unified architecture for generic perception for zero-shot and few-shot tasks. *arXiv:2112.01522*, 2021.

[151] Luisa Zintgraf, Kyriacos Shiarli, Vitaly Kurin, Katja Hofmann, and Shimon Whiteson. Fast context adaptation via meta-learning. In *International Conference on Machine Learning*, 2019.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 5 for a brief discussion and Appendix D for the full discussion.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] The code and the data are proprietary.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 3 and Appendix B.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] We do not observe large enough variance in our training runs to justify the computation cost incurred by multiple training runs. For the largest models, it is not feasible within our compute budget.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Details can be found in Appendix B. In short, our largest run was trained on 1536 TPU chips for 15 days.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes] We properly cited the prior methods on which our work is based, as well as prior datasets when appropriate (e.g., ALIGN).

(b) Did you mention the license of the assets? [N/A] The assets we used are previous work for which we cited papers. We do mention the license of all visual assets we use for the figures of the paper in Appendix G.

(c) Did you include any new assets either in the supplemental material or as a URL? [No]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] Our data was automatically scraped from million of webpages. See Datasheets [29] in Appendix A.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See Datasheets [29] in Appendix A.

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]