Digest

Flamingo

(Paper) Flamingo: a Visual Language Model for Few-Shot Learning

J-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc , A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman, K. Simonyan

arxiv (2022)

Digest by Samuel Albanie, May 2022

Outline

- Motivation and challenges for multimodal generative modelling
- Related Work
- Flamingo model
- Evaluation datasets
- Comparison to state-of-the-art
- Ablations
- Qualitative Examples
- Discussion (limitations, trade-offs, opportunities, benefits, risks)

Motivation

The few-shot dream

Aspect of intelligence: ability to quickly learn a task given short instruction

- Fast acquisition of categories in children (Markman et al., 1989)
- Model learning environment to make better use of data (Griffiths et al. 2019)

We'd like multimodal systems (vision and language) that achieve this property

Dominant computer vision paradigm:

large-scale pretraining + task-specific fine-tuning

But current fine-tuning approaches often require:

- thousands of training samples
- careful per-task hyperparameter tuning
- significant computational resource

Can we train a multimodal model to work well in a "few-shot" regime?

Open-ended task abilities

Multimodal models like CLIP and ALIGN have shown promising zero-shot performance, but they are inflexible: they lack the ability to generate language

Flexible models for visually-conditioned language generation like VL-T5 exist

But these have not demonstrated strong few-shot performance

Inspiration from NLP: large language models like GPT-3 are flexible few-shot learners

Given a few examples of a task as a prompt + query input, the language model generates a continuation to produce a predicted output.

A key factor of their success is large-scale pretraining.

In principle: image/video understanding tasks (e.g. classification, captioning, question-answering) are text prediction problems with visual input conditioning.

Can we we learn a model capable of open-ended multimodal tasks via pretraining?

References

E. Markman, "Categorization and naming in children: Problems of induction", MIT Press (1989)

T. L. Griffiths, et al., "Doing more with less: meta-reasoning and meta-learning in humans and machines", Current Opinion in Behavioral Sciences (2019)

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

(ALIGN) C. Jia et al., "Scaling up visual and vision-language representation learning with noisy text supervision", ICML (2021)

(VL-T5) J. Cho et al., "Unifying vision-and-language tasks via text generation", ICML (2021)

Challenges for multimodal generative modelling

Unifying strong unimodal models

Training large language models is extremely computationally expensive

We'd like to save computational resources by starting from a pretrained language model

But a text-only model has no built-in way to incorporate input from other modalities.

We want to enable this while retaining the knowledge of the original language model

Proposed approach: interleave cross-attention layers with language-only self-attention layers (frozen)

Supporting images and videos

Goal: enable both images and video inputs

These are high-dimensional, so flattening to 1D sequences (as used in text-generation) is costly

Exacerbated by quadratic cost of self-attention

Secondary goal: would also like a unified treatment of images and video

Proposed approach: Perceiver-based architecture with a fixed number of visual tokens

Heterogeneous training data

Large models require vast training datasets.

Existing (image, text) datasets used by (e.g. used by CLIP and ALIGN) my not be general enough to reach GPT-3 style few-shot learning.

Large internet-based text-only datasets exist, but not for multimodal data.

One scalable approach: scrape web pages with interleaved images and text.

But such images and text are often only weakly related

Proposed approach: combine web scraping with existing paired (image, text) and (video, text) datasets

References

A. Jaegle et al., "Perceiver: General perception with iterative attention", ICML (2021)

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

(ALIGN) C. Jia et al., "Scaling up visual and vision-language representation learning with noisy text supervision", ICML (2021)

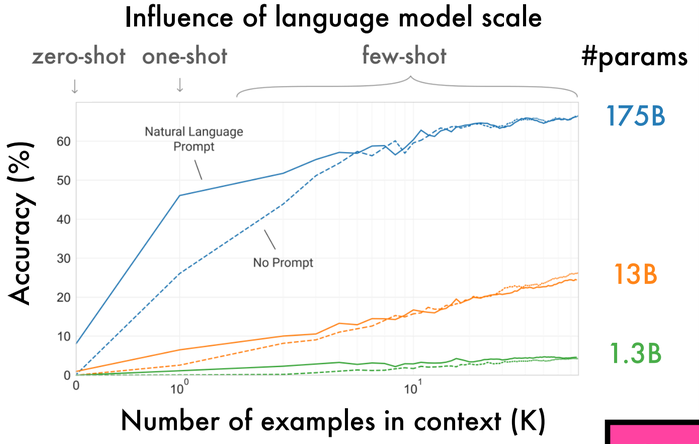
Outline

- Motivation and challenges for multimodal generative modelling
- Related Work
- Flamingo model
- Evaluation datasets
- Comparison to state-of-the-art
- Ablations
- Qualitative Examples
- Discussion (limitations, trade-offs, opportunities, benefits, risks)

Related Work

Brown et al., (2020)

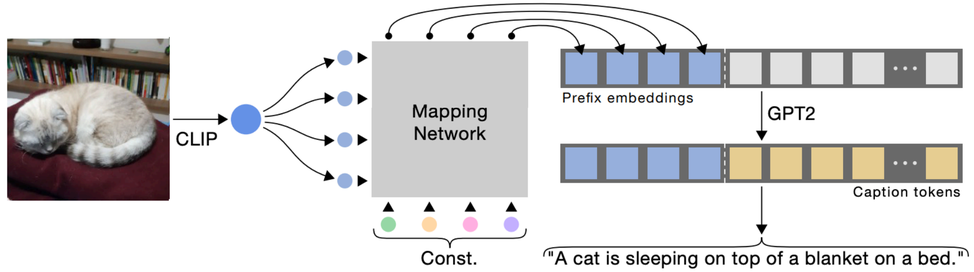
Key idea: large language models are good *in-context* learners (here *in-context learning* refers to learning from text-only examples provided as context)



GPT-3

Mokady et al., (2021)

Key idea: Leverage large trained models (CLIP, GPT-2) for captioning



Achieves solid captioning performance at low computational cost

ClipCap

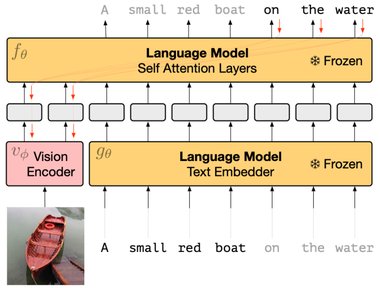
Tsimpoukelli et al., (2021)

Key idea: multimodal few-shot learning with frozen language models via image-conditioned prompt learning

Rapid task adaptation

Encyclopaedic knowledge

Fast binding



Frozen

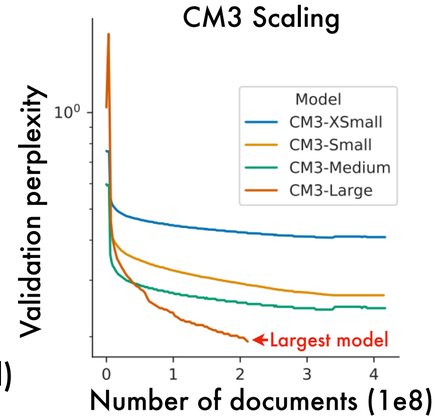
Aghajanyan et al., (2022)

Key idea: combine causal and BERT-style masking on lots of multimodal (HTML) data

	Documents (Millions)	Size (GBs)	Unique Images (Millions)	Tokens (Bllions)
CC-NEWS	45	460	18	121
En-Wikipedia	16	383	7	102
Total	61	843	25	223

Train: 24 days on 384 A100s (large model)

Enables range of unimodal and multimodal tasks



CM3

References/Image credits

T. Brown et al., "Language models are few-shot learners", NeurIPS (2020)

R. Mokady et al., "Clipcap: Clip prefix for image captioning", arxiv (2021)

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

(GPT-2) A. Radford et al., "Language models are unsupervised multitask learners", (2019)

M. Tsimpoukelli et al., "Multimodal few-shot learning with frozen language models", *NeurIPS* (2021)

A. Aghajanyan et al., "CM3: A Causal Masked Multimodal Model of the Internet", arxiv (2022)

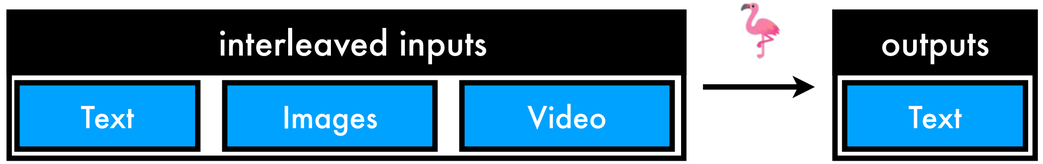
Outline

- Motivation and challenges for multimodal generative modelling
- Related Work
- Flamingo model
- Evaluation datasets
- Comparison to state-of-the-art
- Ablations
- Qualitative Examples
- Discussion (limitations, trade-offs, opportunities, benefits, risks)

Flamingo Model

Approach

Flamingo is a *visual language* model that accepts interleaved inputs:



This enables a broad range of tasks:

open-ended tasks
- visual question answering
- captioning

close-ended tasks
- classification

Goal 1: leverage pretrained models to save compute

Vision: CLIP

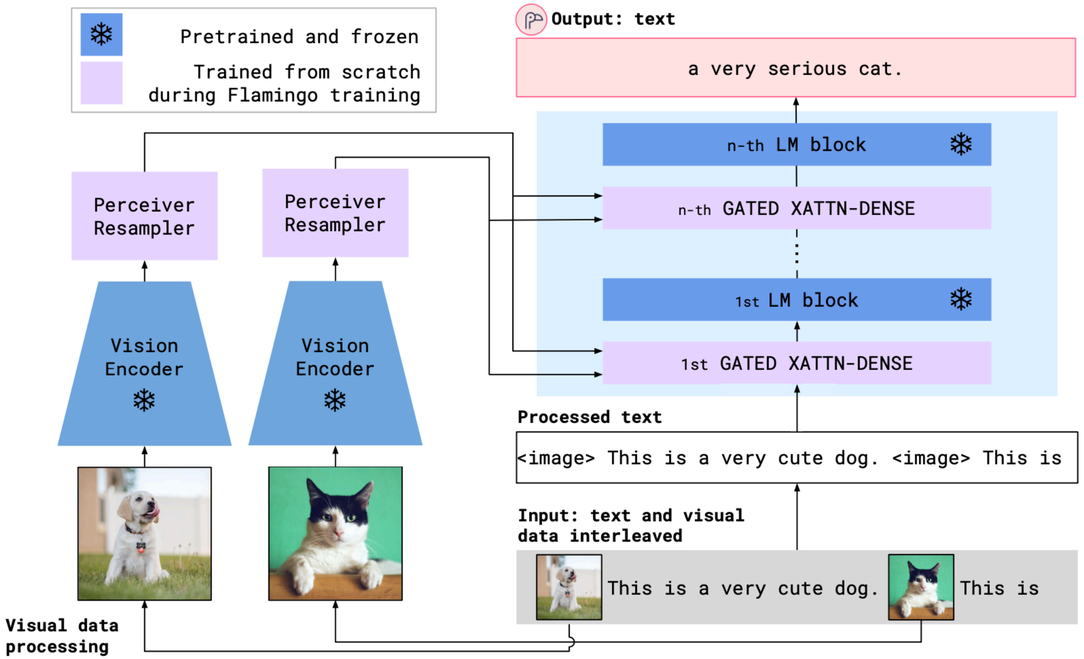
Language: Chinchilla

Goal 2: bridge pretrained models harmoniously

Perceiver resampler

cross-attention

Architecture



Multimodal likelihood

Flamingo models can model the likelihood of text y interleaved with a sequence of images/videos x:

$$p(y \mid x) = \prod_{l=1}^{L} p(y_l \mid y_{<l}, x_{\leq l})$$

$p :=$ flamingo model

$y_l :=$ l-th language token in input text

$y_{<l}$ - preceding text tokens

$x_{\leq l}$ - preceding image/videos

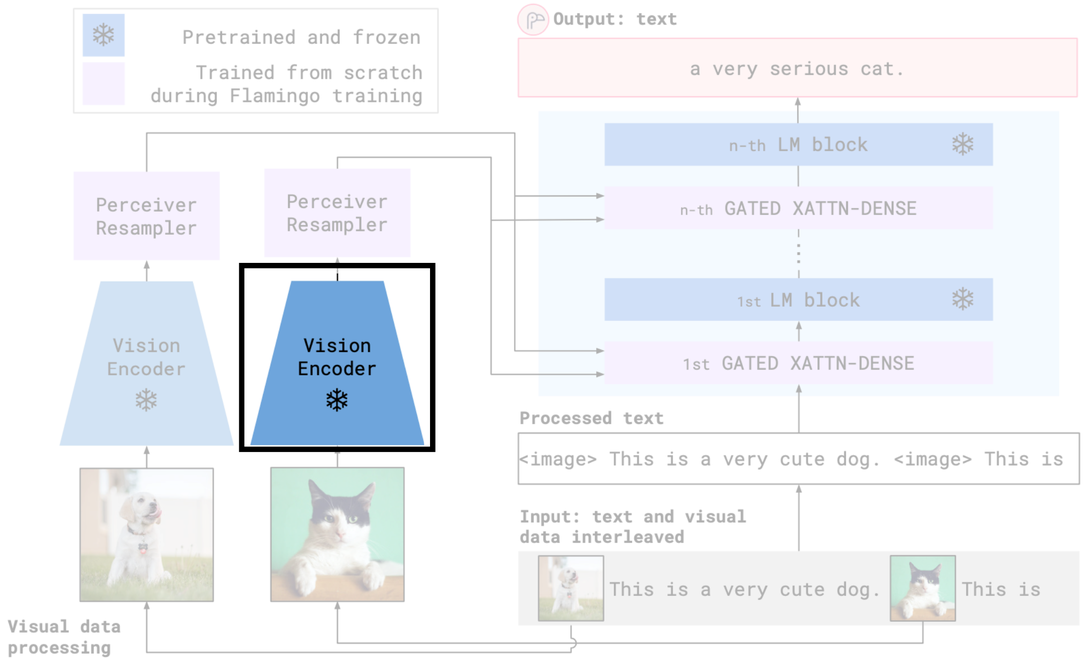
References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

(Chinchilla) J. Hoffmann et al., "Training Compute-Optimal Large Language Models", arxiv (2022)

Vision encoder: pixels to features



Flamingo vision encoder

Vision encoder: F6 Normalizer-Free ResNet (NFNet) backbone

Pretrained as dual encoder using contrastive loss employed by CLIP

BERT is used for the text encoder (discarded after pretraining)

Slight difference to CLIP: global average pooling is used to produce the vision embedding (rather than *global attention pooling*)

Resolution	288 x 288 pixels	Embedding	1376

Outputs 2D spatial grid of features which is flattened to 1D

For videos: frames are sampled at 1FPS (features are concatenated)

Vision encoder is frozen after pretraining

References/Image credits

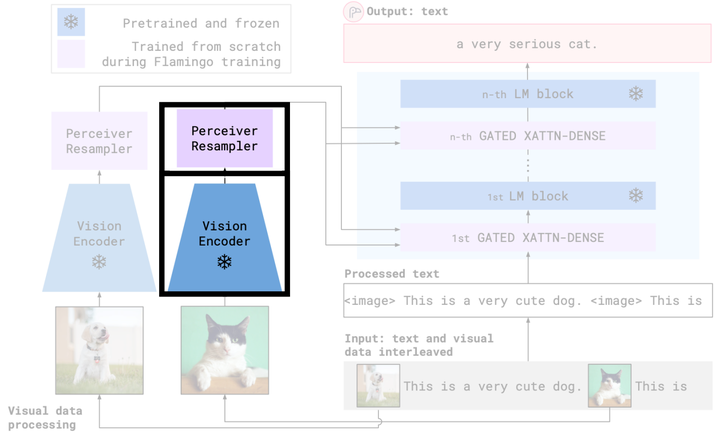
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

(NFNet) A. Brock et al., "High-performance large-scale image recognition without normalization", ICML (2021)

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

(BERT) J. Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", NAACL-HLT (2019)

Vision encoder details



Optimisation details

Trained on 512 TPUv4 chips using Adam optimiser

Batch size of 16,384 (fairly large)

Colour augmentation and random horizontal flips during training

Both the vision encoder and text encoder are trained from scratch

Monitor training progress on zero-shot image classification (like CLIP, this is done with a prompt template "A photo of a {class}")

Pretraining data

Trained on a combination of two internal (image, text) datasets:

ALIGN (1.8 billion) - noisy

LTIP (312 million) - cleaner, longer descriptions

The manner of combination is important for performance

(**Ablation study**) small NFNet-F0 with BERT-mini for different regimes:

Dataset	Combination strategy	ImageNet accuracy top-1	COCO image-to-text R@1	COCO image-to-text R@5	COCO image-to-text R@10	COCO text-to-image R@1	COCO text-to-image R@5	COCO text-to-image R@10
LTIP	None	40.8	38.6	66.4	76.4	31.1	57.4	68.4
ALIGN	None	35.2	32.2	58.9	70.6	23.7	47.7	59.4
LTIP + ALIGN	Accumulation	**45.6**	**42.3**	**68.3**	**78.4**	**31.5**	**58.3**	**69.0**
LTIP + ALIGN	Data merged	38.6	36.9	65.8	76.5	15.2	40.8	55.7
LTIP + ALIGN	Round-robin	41.2	40.1	66.7	77.6	29.2	55.1	66.6

Accumulation: compute gradient on batch from each dataset, combine via weighted sum

Data merged: merge examples from each dataset into each batch

Round-robin: alternate batches from each dataset, update parameters each batch

References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)
(Adam) D. P. Kingma and J. Ba. "Adam: A Method for Stochastic Optimization", ICLR (2015)
(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)
(ALIGN) C. Jia et al., "Scaling up visual and vision-language representation learning with noisy text supervision", ICML (2021)
(NFNet) A. Brock et al., "High-performance large-scale image recognition without normalization", ICML (2021)
(BERT) J. Devlin et al. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", NAACL-HLT (2019)

Perceiver resampler

From large, variable-size features to fixed # tokens

A variable number of input frames are processed (for videos)

The vision encoder thus produces a variable number of features

It outputs a fixed number of visual tokens (64) to limit complexity

Temporal encodings are added to visual inputs (spatial grid position encodings are not use, since they did not help)

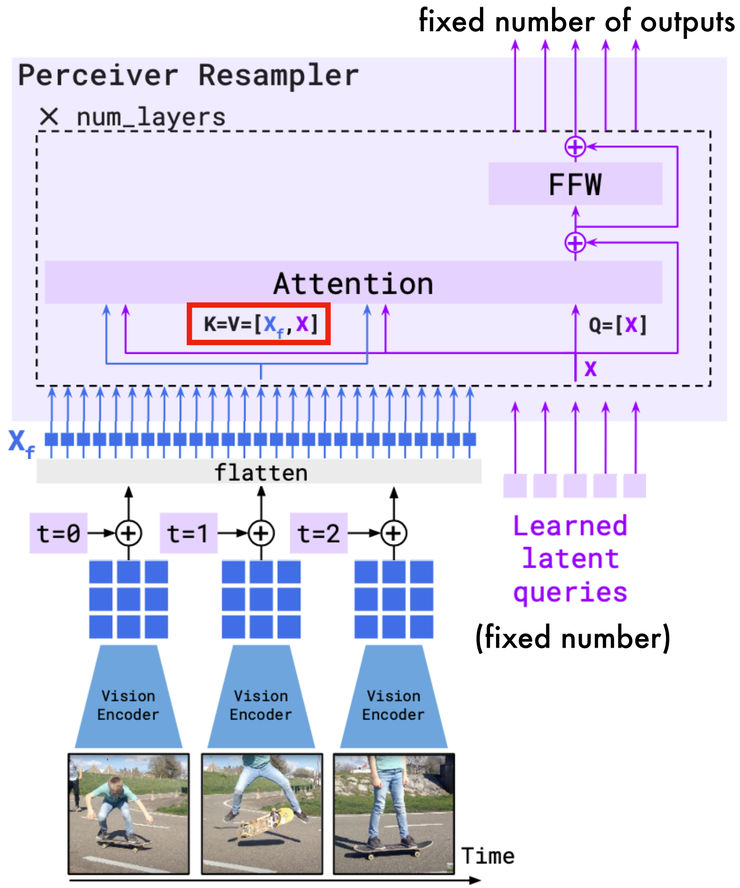
The results are then flattened to form a 1D sequence

These are combined with a fixed set of learned latent queries (64)

Both are processed by attention and feed-forward layers.

Note: differently to DETR and Perceiver, keys and values for latent queries are concatenated to those from the visual embeddings.

Perceiver resampler module



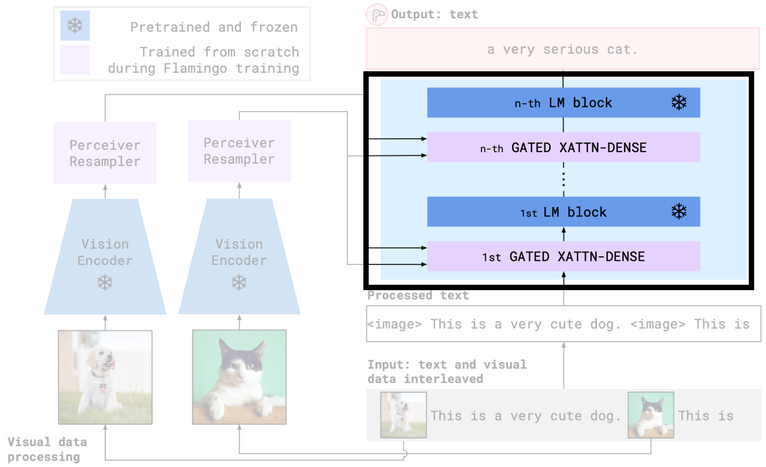
References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

(DETR) N. Carion et al., "End-to-end object detection with transformers", ECCV (2020)

(Perceiver) A. Jaegle et al., "Perceiver: General perception with iterative attention", ICML (2021)

Conditioning the language model



Interleaving gated cross-attention layers

Language models: frozen Chinchillas (trained on MassiveText)

Gated xattn dense blocks (trained from scratch) are inserted between layers

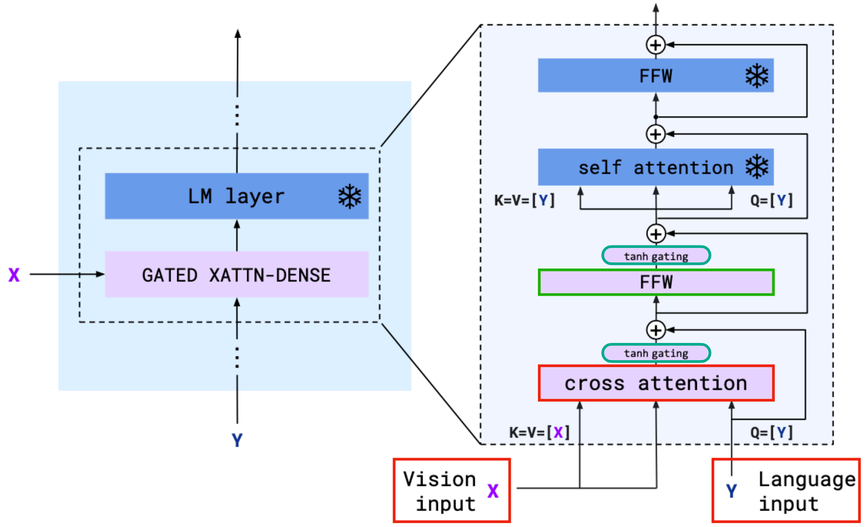
Each block includes cross attention feed-forward

Layer norm is applied to all attention inputs and the feed-forward layers (GPT-2 style)

Use tanh gates to preserve original language model behaviour at initialisation

Each $\tanh(\alpha)$ gate controlled via a layer-specific learnable scalar α (initialised to zero)

Gated xattn-dense block structure



Architecture integration

Gated xattn-dense insertions are chosen according to the selected language model

Differing insertion frequencies represent different compute-performance trade-offs

References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)
(MassiveText) J. Rae, et al. "Scaling language models: Methods, analysis & insights from training gopher", arxiv (2021)
(Chinchilla) J. Hoffmann et al., "Training Compute-Optimal Large Language Models", arxiv (2022)
(Layer norm) J. Ba et al., "Layer normalization", arxiv (2016)
(GPT-2) A. Radford et al., "Language models are unsupervised multitask learners" (2019)
(tanh gates) S. Hochreiter et al., "Long short-term memory", *Neural computation* (1997)

Per-image/video attention masking

Interleaved sequences

Each example sequence consists of:

text y | images/videoes x

positions of images/videos in text

Define function $\Phi : [1, L] \rightarrow [0, N]$

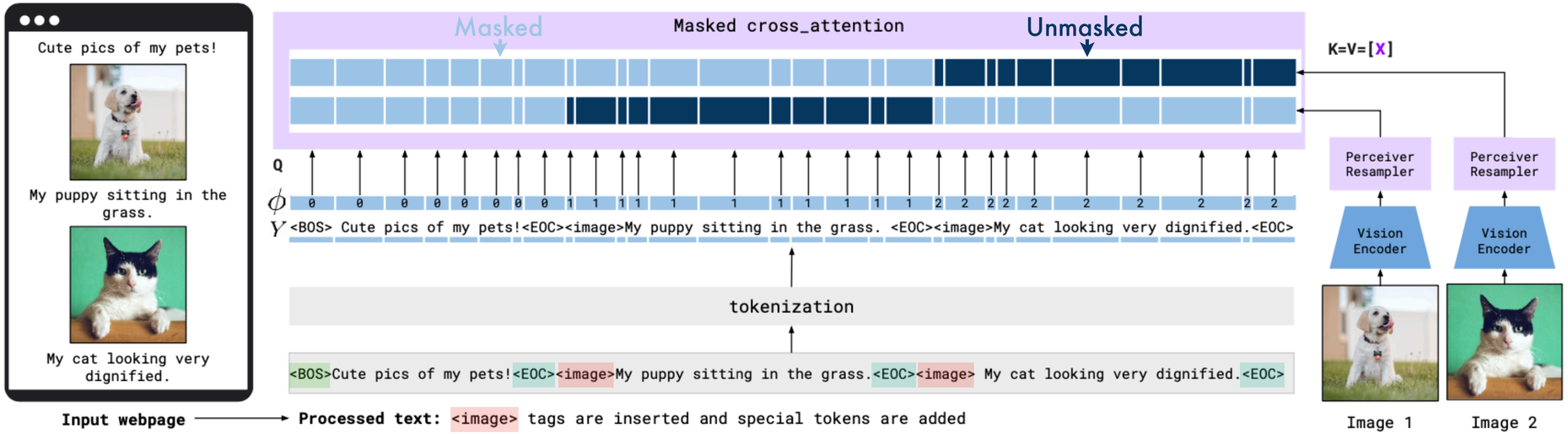
This assigns to each text position the index of the last image/video (or 0).

Φ indicates which visual inputs can be used to predict token l:

$y_{<l} \triangleq (y_1, \ldots, y_{l-1})$

$x_{\leq l} \triangleq (x_i \mid i \leq \phi(l))$

Masked cross attention



Multi-image attention

Multi-image attention is implemented with the gated xattn-dense layers with causal masking over tokens from the perceiver resampler.

By default, each token only allowed to attend to the visual tokens of the image that appeared immediately before it (this restriction improved performance)

Note: Although direct attention is over a single image, there is still a causal dependency on previous images (due to causal self-attention in the text decoder)

Experiments show that the model can train on 5 images, but generalise up to 32. Restriction may be a useful inductive bias for single image tasks.

References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Flamingo - training data

Data sources

Flamingo is trained on:

- Image-Text Pairs data
- Video-Text Pairs data
- Webpage data

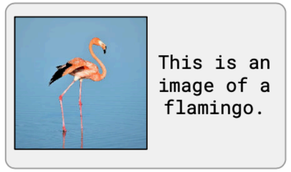


Image-Text Pairs dataset
[N=1, T=1, H, W, C]

Video-Text Pairs dataset
[N=1, T>1, H, W, C]

Multi-Modal Massive Web (M3W) dataset
[N>1, T=1, H, W, C]

Image-Text Pairs data

ALIGN dataset

- 1.8 billion noisy image-text pairs
- 12.4 text tokens per image on average

LTIP dataset

- 312 million image-text pairs
- 20.5 text tokens per image on average

Resolution	320 x 320 pixels	Text	$L = 32/64$

Video-Text Pairs data

VTP dataset

- 27 million short videos
- 22 seconds in duration on average

Resolution	320 x 320 pixels
Temporal dim	T = 8
Text	$L = 32$

MultiModal Massive Web (M3W)

Extract text and images from 43 million webpages

Use the DOM to determine the interleaving order of text and images

M3W contains 185 million images and 182 GB of text

Documents are filtered with Google SafeSearch filter

Text filters: heuristics used to remove low quality documents & repetitions

Image filters: small (< 64 pixels), extreme aspect ratio, single-colour

Resolution	320 x 320 pixels	Text	$L = 256$

References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)
(ALIGN) C. Jia et al., "Scaling up visual and vision-language representation learning with noisy text supervision", ICML (2021)
(SafeSearch) https://support.google.com/websearch/answer/510
(Datasheets) T. Gebru et al., "Datasheets for datasets", Communications of the ACM (2021)

(Datasheets for LTIP, VTP, M3W)

Train/val splits are randomly chosen

internal datasets

Flamingo training objective

Training details

Models are trained with a weighted sum of dataset specific negative log likelihoods of text (conditioned on visual inputs):

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

$\mathcal{D}_m$ - m-th dataset

λ_m - positive scalar weight for the m-th dataset

Similarly to the vision encoder pretraining:

- tuning these weights is important for good performance
- the accumulation strategy is used

Task adaptation with few-shot in-context learning

In-context learning - building multimodal prompts

Evaluate ability to rapidly adapt to new tasks in the style of GPT-3

Input format: support examples (image, text) or (video, text) + visual query

Vision to Text tasks (input=vision, output=text)

Support examples

A cat wearing sunglasses.

Elephants walking in the savanna.

Query

Output:

```
<BOS><image>Output: A cat wearing sunglasses.<EOC><image>Output: Elephants walking in the savanna.<EOC><image>Output:
```

Processed prompt

Visual Question Answering Task (input=vision+text, output=text)

Support examples

What's the cat wearing?

sunglasses

How many animals?

3

Query

What is on the water?

Question:

Answer:

```
<BOS><image>Question: What's the cat wearing? Answer: sunglasses<EOC><image>Question: How many animals? Answer: 3<EOC>
<image>Question: What is on the water? Answer:
```

Processed prompt

Note: by default, the support examples are provided in random order.

References/Image credits

(GPT-3) T. Brown et al., "Language models are few-shot learners", NeurIPS (2020)

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Few-shot in-context learning details

Open-ended and close-ended evaluations

Open-ended tasks

- the text generated following the query image is used as the prediction
- stop at the first <EOC> token
- beam search with beam size 3 is used

Close-ended tasks

- each target candidate is appended independently to the query image
- sequences are ranked by their log-likelihood

Zero-shot generalisation

If no examples available, one option is prompt engineering (CLIP)

Performance is sensitive to the prompt, but validation requires examples

Perez et al. (2021) shows that validation with few samples is not robust

Flamingo: Build prompt from two downstream examples without images/video

Using one example worked poorly (model predictions are very similar to the example)

Min et al., (2022): (label space/text distribution/format matter, label correctness doesn't)

For close-ended tasks, no text examples are required for the zero-shot prompt.

Retrieval-based in-context example selection

It can be hard to leverage large numbers of support examples:

- it is expensive to fit all examples in the prompt
- generalisation may suffer if fewer examples were used in training

Prompt example selection (Liu et al., 2021) can address these issues

Retrieval-based In-Context Example Selection (RICES) (Yang et al., 2021)

Build prompt from top-N most similar examples to query

To avoid recency bias (Zhao et al., 2021), most similar example is put last

Prompt ensembling

Ensembling across multiple prompts can be used to improve performance

Note: This can be combined with RICES over different permutations of nearest neighbours

For a given answer, log likelihoods are ensembled over six random permutations of the selected few-shot prompts

References/Image credits

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

E. Perez et al., "True few-shot learning with language models", NeurIPS (2021)

S. Min et al., "Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?", arxiv (2022)

J. Liu et al., "What Makes Good In-Context Examples for GPT-$3?", arxiv (2021)

(RICES) Z. Yang et al., "An empirical study of gpt-3 for few-shot knowledge-based vqa", arxiv (2021)

Z. Zhao et al., "Calibrate before use: Improving few-shot performance of language models", ICML (2021)

Flamingo Models

Model architectures

Three sizes of Flamingo model are considered (building on three Chinchilla sizes)

	Requires model sharding	Frozen Language	Frozen Vision	Trainable GATED XATTN-DENSE	Trainable Resampler	Total count
Flamingo-3B	✗	1.4B	435M	1.2B (every)	194M	**3.2B**
Flamingo-9B	✗	7.1B	435M	1.6B (every 4th)	194M	**9.3B**
Flamingo	✓	70B	435M	10B (every 7th)	194M	**80B**

The largest (80B) model requires model sharding

All models use a NFNet-F6 backbone for the frozen vision encoder

Gated xattn-dense layers are inserted at different frequencies (trading off memory and performance)

The Perceiver Resampler remains the same across each model

Model card: model intended for internal development.

References/Image credits
(Chinchilla) J. Hoffmann et al., "Training Compute-Optimal Large Language Models", arxiv (2022)
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)
(NFNet) A. Brock et al., "High-performance large-scale image recognition without normalization", ICML (2021)
(Perceiver) A. Jaegle et al., "Perceiver: General perception with iterative attention", ICML (2021)
(Model card) M. Mitchell et al., "Model cards for model reporting", ACM FAccT (2019)

Outline

- Motivation and challenges for multimodal generative modelling
- Related Work
- Flamingo model
- Evaluation datasets
- Comparison to state-of-the-art
- Ablations
- Qualitative Examples
- Discussion (limitations, trade-offs, opportunities, benefits, risks)

Few-shot evaluation benchmarks

Benchmark datasets

image benchmarks

ImageNet-1K	object classification	unimodal
MS-COCO	scene description	generative
VQAv2	scene understanding QA	generative
OKVQA	external knowledge QA	generative
Flickr30k	scene description	generative
VizWiz	scene understanding QA	generative
TextVQA	text reading QA	generative
VisDial	visual dialogue	
HatefulMemes	meme classification	custom prompt

video benchmarks

Kinetics700	action classification	unimodal
VATEX	event description	generative
MSVDQA	event understanding QA	generative
YouCook2	event description	generative
MSRVTTQA	event understanding QA	generative
iVQA	event understanding QA	generative
RareAct	composite action retrieval	custom prompt
NextQA	temporal/causal QA	generative
STAR	multiple choice QA	

2 datasets are unimodal

12 benchmarks require open-ended generative sampling

7 "DEV" benchmarks are used to guide development

splits: val support | val query | test support | test query

11 used for a less biased few-shot evaluation (includes less explored capabilities):

OCR | visual dialogue | vision and text classification

compositional actions | causal relations | multi-choice QA

custom prompts

HatefulMemes	"is an image with written: "{meme_text}" on it. Is it hateful? Answer: {answer}"
RareAct	"Caption: a person {verb + object}"

Flamingo: dataset deduplication

Data deduplication against evaluation tasks

Flamingo uses large-scale web-based pretraining

Necessary investigate the possibility of evaluation dataset contamination

CLIP - did not deduplicate (instead performed analysis)

ALIGN did perform deduplication

Flamingo uses an internal Google tool for deduplication

Nearest neighbour search via visual embeddings to retrieve duplicates

Deduplication details

(Image-text) Deduplicated LTIP and ALIGN training images against:

- ImageNet (train, val)
- COCO (train, val, test)
- OK-VQA (train, val, test)
- VQAv2 (train, val, test)
- Flickr30k (train, valid, test)
- VisDial (train, valid, test)

Did not deduplicate against:

- VizWiz
- HatefulMemes
- TextVQA

Authors suggest this did not impact results (images are unlikely to be scraped from web)

Did not deduplicate M3W (unlikely to contain images from benchmark suite)

Verification stats: 1314 potential duplicates between six datasets above and M3W

125 of these are exact duplicates

(Video-text) Did not deduplicate VTP

No videos were collected from YouTube/Flickr (source of all video evaluation datasets)

References
(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)
(ALIGN) C. Jia et al., "Scaling up visual and vision-language representation learning with noisy text supervision", ICML (2021)

Flamingo: nuts and bolts training details

Data augmentation and pre-processing

During training, 50% of text samples are prepended with a space character

Effectiveness likely due subword tokenizer (tokens depend on preceding space)

Augmentation enforces invariance to this artifact without degrading punctuation

Visual inputs processed at 320 pixels (rather than 288 pixels used in pretraining)

Inspired by FixRes - not too expensive since vision encoder is frozen

On interleaved datasets, image indices ϕ are also perturbed (next/prev prob. 0.5)

For videos: clips of 8 frames (at 1 fps) are sampled from each training video

In inference: 30 video frames processed at 3 fps (interpolating pos. embeddings)

Loss and optimisation

All models are trained with AdamW

Optimisation is done with linear warmup followed by a flat learning rate

Dataset mixing weights (λ_m):

M3W	1.0	**LTIP**	0.2
Align	0.2	**VTP**	0.03

Infrastructure/implementation

The model is trained using JAX and Haiku

Training used TPUv4 instances

Largest (80B) model trained for 15 days for 1536 chips over 16 devices

Megatron sharding used for Embedding/S-Attention/X-Attenion/FFW

ZeRO stage 1 is used to shard the optimiser state

Activations + gradients: bfloat16, params + optim. accumulators: float32

References

H. Touvron et al., "Fixing the train-test resolution discrepancy: FixEfficientNet", arxiv (2020)

D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", *ICLR* (2015)

I. Loshchilov and F. Hutter, "Decoupled weight decay regularization", arxiv (2017)

(JAX) J. Bradbury et al., "JAX: composable transformations of Python+ NumPy programs" (2018)

(Haiku) T. Hennigan et al., "Haiku: Sonnet for Jax" (2020)

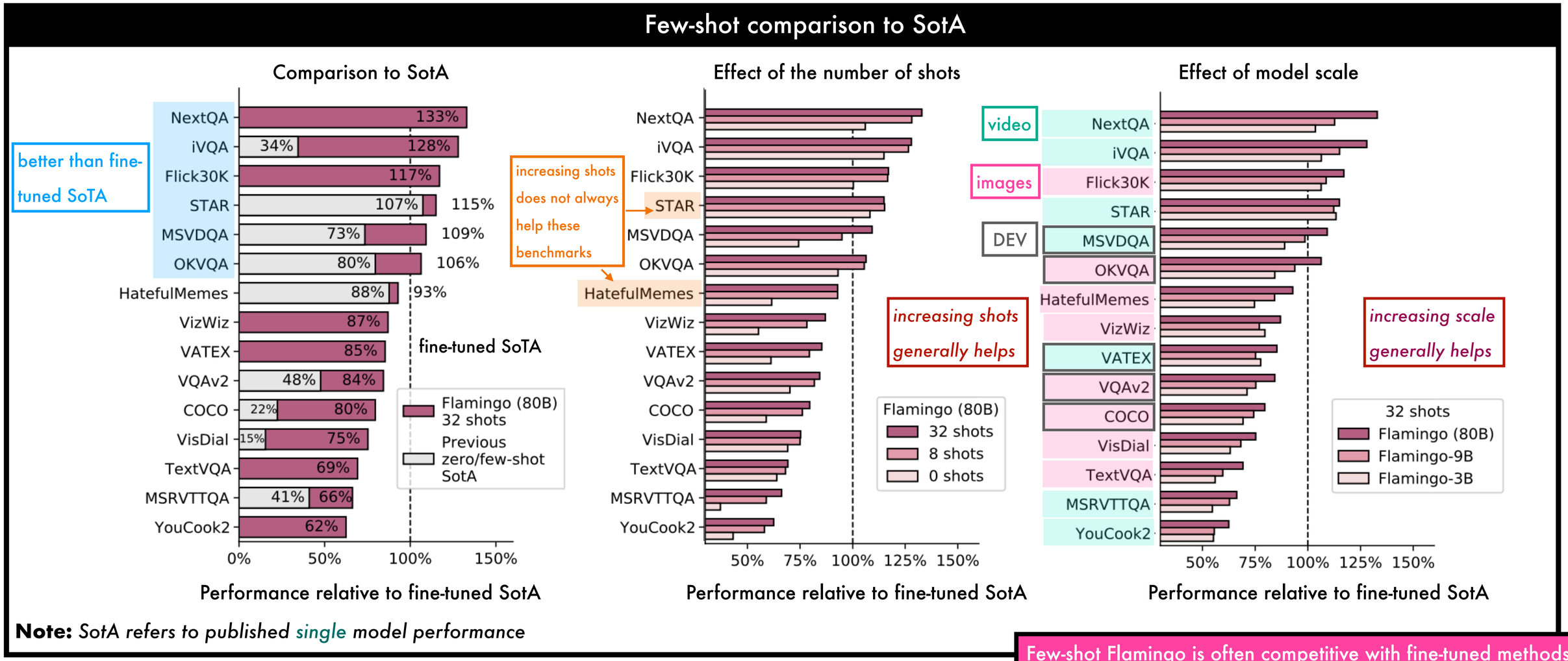
(Megatron) M. Shoeybi et al., "Megatron-lm: Training multi-billion parameter language models using model parallelism", arxiv (2019)

(ZeRO) S. Rajbhandari et al., "Zero: Memory optimizations toward training trillion parameter models", SC (2020)

Outline

- Motivation and challenges for multimodal generative modelling
- Related Work
- Flamingo model
- Evaluation datasets
- Comparison to state-of-the-art
- Ablations
- Qualitative Examples
- Discussion (limitations, trade-offs, opportunities, benefits, risks)

Few-shot: comparison to SotA

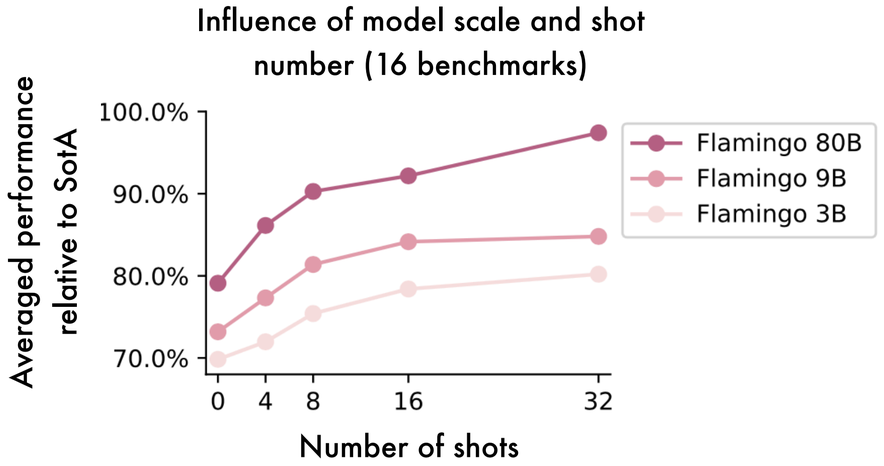


References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Few-shot: further analysis

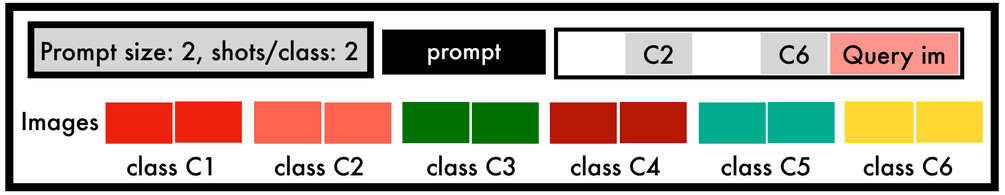
Model scaling and shot number



Additional shots improve performance (like GPT-3)

The largest model better leverages increases in shot number

Note: flamingo trained with sequences of ≤5 images on M3W



Few-shot classification on classification tasks

Model	Method	Prompt size	shots/class	ImageNet top 1	Kinetics700 avg top1/5
SotA	Fine-tuned	-	full	91.0 soup	89.0 MTV
SotA	Contrastive	-	0	**85.7** BASIC	**69.6** CLIP
NFNetF6	Our contrastive	-	0	77.9	62.9
Flamingo-3B	RICES	8	1	70.9	55.9
		16	1	71.0	56.9
		16	5	72.7	58.3
Flamingo-9B	RICES	8	1	71.2	58.0
		16	1	71.7	59.4
		16	5	75.2	60.9
Flamingo-80B	Random	16	≤ 0.02	66.4	51.2
	RICES	8	1	71.9	60.4
		16	1	71.7	62.7
		16	5	76.0	63.5
	RICES+ensembling	16	5	77.3	64.2

Fine-tuned SotA soup: ViT-G/14 (model soup) MTV: Multiview Transformer for Video Recognition

Zero-shot SotA BASIC: 3B params, 6.6B image-text pairs CLIP: ViT-L/14@336 px

References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

(GPT-3) T. Brown et al., "Language models are few-shot learners", NeurIPS (2020)

(Model soup) M. Wortsman et al., "Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time", arxiv (2022)

(MTV) S. Yan et al., "Multiview Transformers for Video Recognition", arxiv (2022)

(BASIC) Pham et al., "Combined Scaling for Zero-shot Transfer Learning", arxiv (2021)

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

Contrastive pretraining: zero-shot retrieval

Retrieval benchmarks

Evaluation of the pretrained dual encoder for zero-shot retrieval

	Flickr30K						COCO					
	image-to-text			text-to-image			image-to-text			text-to-image		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Florence	**90.9**	**99.1**	-	76.7	93.6	-	64.7	85.9	-	47.2	71.4	-
ALIGN	88.6	98.7	**99.7**	75.7	93.8	96.8	58.6	83.0	89.7	45.6	69.8	78.6
CLIP	88.0	98.7	99.4	68.7	90.6	95.2	58.4	81.5	88.1	37.7	62.4	72.2
Flamingo	89.3	98.8	**99.7**	**79.5**	**95.3**	**97.9**	**65.9**	**87.3**	**92.9**	**48.0**	**73.3**	**82.1**

Observation: training on short text descriptions improves ImageNet classification but harms text-image retrieval

Flamingo pretraining optimises for retrieval rather than classification to capture the whole scene in images.

References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

(Florence) L. Yuan et al., "Florence: A New Foundation Model for Computer Vision", arxiv (2021)

(ALIGN) C. Jia et al., "Scaling up visual and vision-language representation learning with noisy text supervision", ICML (2021)

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

Fine-tuning Flamingo

Low-level details

During fine-tuning, Flamingo keeps the language model layers frozen

Input image resolution is increased from 320 x 320 pixels to 480 x 480 pixels

The base vision encoder is also fine-tuned (unlike Flamingo pretraining)

Hyperparameters are set by grid search on validation subsets of the training sets

Search over: learning rate, decay schedule, training steps, batch size, augmentation

Fine-tuning comparison on 9 benchmarks

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
Flamingo - 32 shots	67.6	-	113.8	65.1	49.8	-	31.0	56.8	-	86.8	36.0	-	70.0
SimVLM	80.0	80.3	**143.3**	-	-	-	-	-	-	-	-	-	-
OFA	79.9	80.0	149.6	-	-	-	-	-	-	-	-	-	-
Florence	80.2	80.4	-	-	-	-	-	-	-	-	-	-	-
Flamingo 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**
Restricted SotA†	80.2 [150]	80.4 [150]	**143.3** [134]	76.3 [165]	- -	- -	46.8 [57]	**75.2** [87]	**74.5** [87]	**138.7** [142]	54.7 [147]	**73.7** [92]	75.4 [60]
Unrestricted SotA	81.3 [143]	81.3 [143]	149.6 [129]	81.4 [165]	57.2 [70]	60.6 [70]	- -	- -	75.4 [133]	- -	- -	- -	84.6 [164]

Summary: fine-tuning, while expensive, brings significant gains in performance

References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

(SimVLM) Z. Wang et al., "Simvlm: Simple visual language model pretraining with weak supervision", arxiv (2021)

(OFA) P. Wang et al., "Unifying Architectures, Tasks, and Modalities Through a Simple Sequence-to-Sequence Learning Framework", arxiv (2022)

(Florence) L. Yuan et al., "Florence: A New Foundation Model for Computer Vision", arxiv (2021)

Outline

- Motivation and challenges for multimodal generative modelling
- Related Work
- Flamingo model
- Evaluation datasets
- Comparison to state-of-the-art
- Ablations
- Qualitative Examples
- Discussion (limitations, trade-offs, opportunities, benefits, risks)

Ablation studies

Low-level details

Ablations use the Flamingo-3B model

Metrics reported on 7 validation DEV datasets

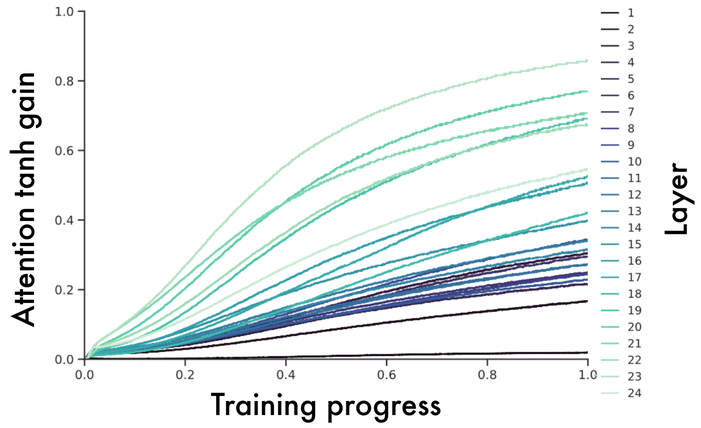
Each experiment uses 4 shots

Batch sizes:

M3W	256	**ALIGN**	512
LTIP	256	**VTP**	64

Models are trained for 1 million gradient steps

Evolution of attention tanh gating



Influence of training data mixture

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	**68.4**
(i)	Training data	All data	M3W	3.2B	0.68s	58.0	37.2	48.6	35.7	29.5	33.6	34.0	50.7
			w/o VTP	3.2B	1.42s	84.2	43.0	53.9	59.6	34.5	46.0	45.8	65.4
			w/o LTIP/ALIGN	3.2B	0.95s	66.3	39.2	51.6	41.4	32.0	41.6	38.2	56.5
			w/o M3W	3.2B	1.02s	54.1	36.5	52.7	24.9	31.4	23.5	28.3	46.9

For some dataset combinations, no <EOC> token is produced (instead additional prompts are predicted)

For these cases, the prediction is trimmed to the text preceding the prompt keywords.

Optimisation strategy for mixing datasets

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	**68.4**
(ii)	Optimisation	Grad. accumulation	Round Robin	3.2B	1.68s	76.1	39.8	52.1	50.7	33.2	40.8	39.7	59.7

Architecture: tanh cross-attention gating

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	**68.4**
(iii)	Tanh gating	✓	✗	3.2B	1.74s	78.4	40.5	52.9	54.0	35.9	47.5	46.4	64.0

Conditioning architectures for the frozen language model

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	**68.4**
(iv)	Cross-attention architecture	GATED XATTN-DENSE	VANILLA XATTN	2.4B	1.16s	80.6	41.5	53.4	59.0	32.9	50.7	46.8	65.2
			GRAFTING	3.3B	1.74s	79.2	36.1	50.8	47.5	32.2	47.8	27.9	57.4

References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

(Round Robin) J. Cho et al., "Unifying vision-and-language tasks via text generation", ICML (2021)

(Vanilla XAttn) A. Vaswani et al., "Attention is all you need", NeurIPS (2017)

(Grafting) Z. Luo et al., "VC-GPT: Visual Conditioned GPT for End-to-End Generative Vision-and-Language Pre-training", arxiv (2022)

Ablation studies cont.

Compute/capacity vs. performance trade-off for cross-attention

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	68.4
(v)	Cross-attention frequency	Every	Single in middle	2.0B	0.87s	71.5	38.1	50.2	44.0	29.1	42.3	28.3	54.6
			Every 4th	2.3B	1.02s	82.3	42.7	55.1	57.1	34.6	50.8	45.5	65.9
			Every 2nd	2.6B	1.24s	83.7	41.0	55.8	59.6	34.5	49.7	47.4	66.2

Resampler architecture and size

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	68.4
(vi)	Resampler	Perceiver	MLP	3.2B	1.85s	78.6	42.2	54.7	53.6	35.2	44.7	42.1	63.3
			Transformer	3.2B	1.81s	83.2	41.7	55.6	59.0	31.5	48.3	47.4	65.1
(vii)	Resampler size	Medium	Small	3.1B	1.58s	81.1	40.4	54.1	60.2	36.0	50.2	48.9	66.4
			Large	3.4B	1.87s	84.4	42.2	54.4	60.4	35.1	51.4	49.4	67.3

Number of images attended to

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	68.4
(viii)	Multi-Img att.	Only last	All previous	3.2B	1.74s	70.0	40.9	52.0	52.3	32.1	46.8	42.0	60.8

M3W image placement data augmentation

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	68.4
(ix)	p_{next}	0.5	0.0	3.2B	1.74s	85.0	41.6	55.2	60.3	36.7	50.6	49.9	67.8
			1.0	3.2B	1.74s	81.3	43.3	55.6	57.8	36.8	52.7	47.8	67.6

References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Ablation studies cont.

Vision encoder pretraining

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	68.4
(x)	Vision encoder	NFNet-F6	CLIP ViT-L/14	3.1B	1.58s	76.5	41.6	53.4	49.5	33.2	44.5	42.3	61.4
			NFNet-F0	2.9B	1.45s	73.8	40.5	52.8	49.8	31.1	42.9	36.6	58.9

Language model pretraining

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	68.4
(xi)	LM pretraining	MassiveText	C4	3.2B	1.74s	81.3	34.4	47.1	60.6	30.9	53.9	46.9	62.5

Freezing model components to prevent catastrophic forgetting

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	68.4
(xii)	Freezing Vision	✓	✗ (random init)	3.2B	4.70s*	74.5	41.6	52.7	45.2	31.4	35.8	32.6	56.6
			✗ (pretrained)	3.2B	4.70s*	83.5	40.6	55.1	55.6	34.6	50.7	41.2	64.5
(xiii)	Freezing LM	✓	✗ (random init)	3.2B	2.42s	74.8	31.5	45.6	59.5	26.9	50.1	43.4	58.2
			✗ (pretrained)	3.2B	2.42s	81.2	33.7	47.4	60.7	31.0	53.9	49.9	62.9

Co-training the language model on MassiveText

	Ablated setting	Flamingo 3B value	Changed value	Param. count ↓	Step time ↓	COCO CIDEr↑	OKVQA top1↑	VQAv2 top1↑	ImageNet top1↑	MSVDQA top1↑	VATEX CIDEr↑	Kinetics top1-top5↑	Overall score↑
	Flamingo 3B model (short training)			3.2B	1.74s	86.5	42.1	55.8	59.9	36.3	53.4	49.4	68.4
(xiv)	Co-train LM on MassiveText	✗	✓ (random init)	3.2B	5.34s*	69.3	29.9	46.1	59.9	28.1	45.5	46.9	57.4
			✓ (pretrained)	3.2B	5.34s*	83.0	42.5	53.3	60.9	35.1	51.1	50.1	67.2

MassiveText is added to the mixture with $\lambda_m = 1.0$ (determined by grid search)

References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

(MassiveText) J. Rae, et al. "Scaling language models: Methods, analysis & insights from training gopher", arxiv (2021)

C. Raffel, et al., "Exploring the limits of transfer learning with a unified text-to-text transformer", JMLR (2019)

Outline

- Motivation and challenges for multimodal generative modelling
- Related Work
- Flamingo model
- Evaluation datasets
- Comparison to state-of-the-art
- Ablations
- Qualitative Examples
- Discussion (limitations, trade-offs, opportunities, benefits, risks)

Qualitative results

Low-level details

To generate qualitative examples, greedy decoding is used, rather than beam search (as used for benchmarks)

The examples are selected (rather than random)

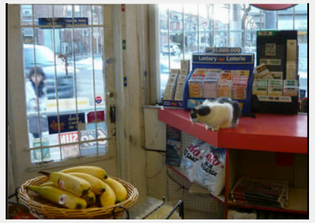
Text completion of image and text prompts

Input Prompt	 Question: What do you think the capacities of these are? Answer:	 Question: What is odd about this image? Explain why it is unusual. Answer:	 Question: What country is this? Why do you think so? Answer:
Completion	**The floppy disk is 1.44MB and the CD is 700MB.**	**The image is odd because the elephant is in the back of the truck. It is unusual because elephants are not usually transported in the back of a truck.**	**It is Canada. I think so because the flag is the Canadian flag.**

References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

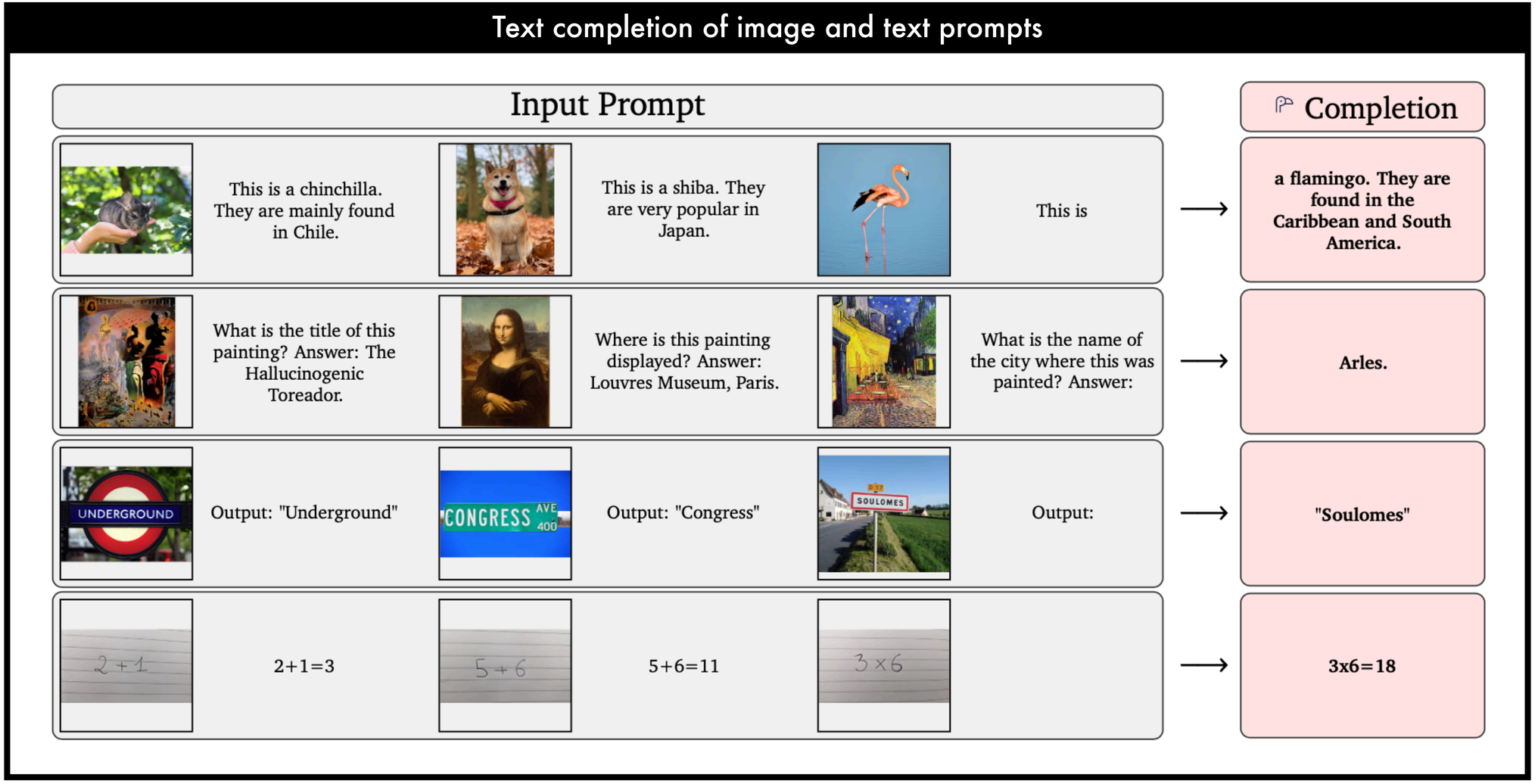
Qualitative results

Text completion of image and text prompts

Input Prompt	 Question: What nutrients is the fruit in the image rich in? Answer:	 Question: What is the purpose of this? Answer:	 Question: What season is this? Explain why. Answer:	 Question: What is the title of the book? Answer:	 The ingredients of this dish are	 The soundtrack includes
Completion	**Vitamin C, potassium, and fiber.**	**To keep the mosquitoes out of the bed.**	**This is winter season. The Christmas tree and the presents are the main reasons why it is winter.**	**The House Book.**	**: beef, potatoes, carrots, broccoli, and lemon.**	**the songs "Let It Go" and "For the First Time in Forever" by Tony Award® winner Idina Menzel, who plays Elsa [...]**

References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

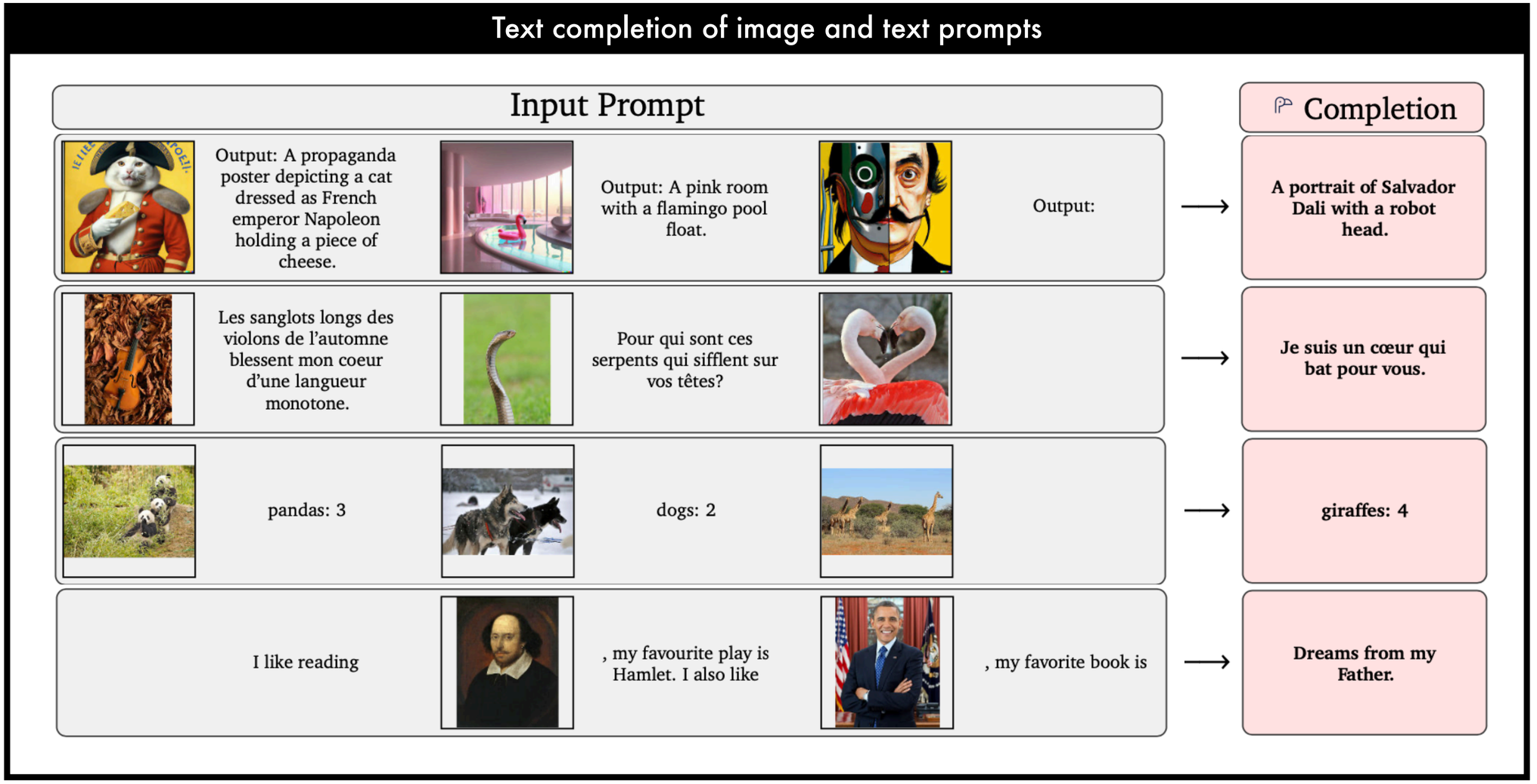
Qualitative results



References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Qualitative results



References/Image credits

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Qualitative results - dialogue

Prompt

This is a conversation between a human, User, and an intelligent visual AI, Flamingo. User sends images, and Flamingo describes them.

User: <a cat image>

Flamingo: That is a cat. It's a tiny kitten with really cute big ears.

User: <a dinner image>

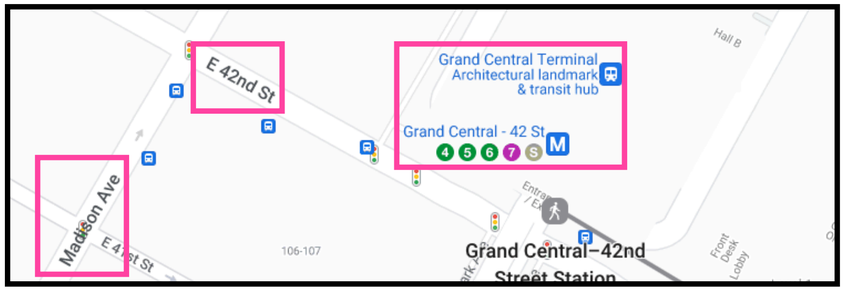
Flamingo: This is a picture of a group of people having dinner. They are having a great time!

User: Can you guess what are they celebrating?

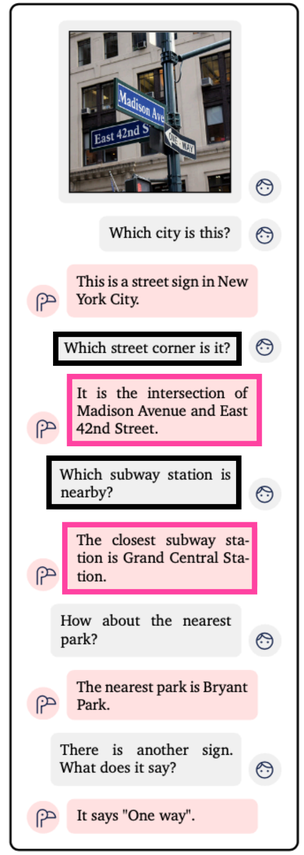
Flamingo: They might be celebrating the end of a successful project or maybe a birthday?

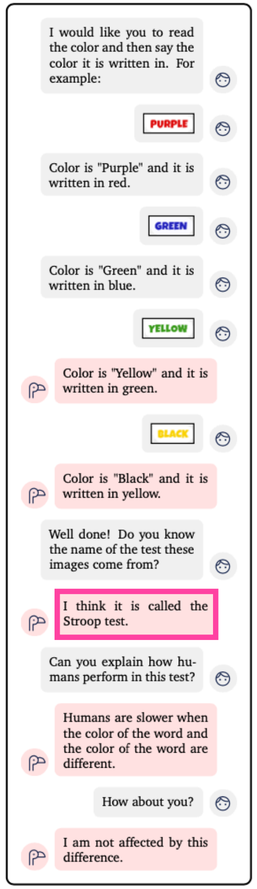
User: <a graph image>

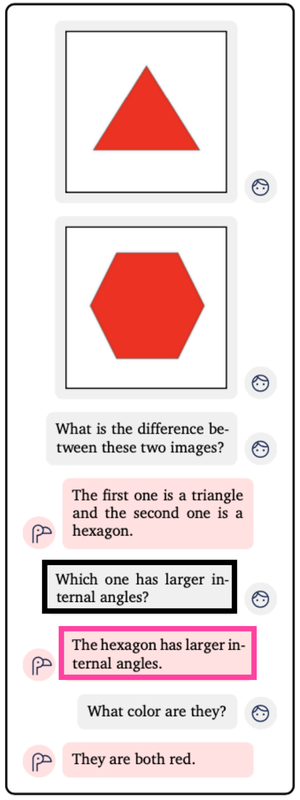
Flamingo: This is a graph, it looks like a cumulative density function graph.



Selected dialogue samples







References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)
https://www.rome2rio.com/map/Grand-Central-42nd-Street-Station/590-Madison-Avenue

Qualitative results - dialogue

Additional selected dialogue samples

What is in this picture?

It's a bowl of soup with a monster face on it.

What is the monster made out of?

It's made out of vegetables.

No, it's made out of a kind of fabric. Can you see what kind?

It's made out of a woolen fabric.

This is a picture of a dog. It looks like it is a little bit naughty.

What happened here?

It looks like the dog is destroying the sofa.

How does the person thinking about it?

The person is angry.

What is the person holding?

It looks like a cup of coffee.

How is the weather outside?

It looks like a sunny day.

What is the name of the company on this card?

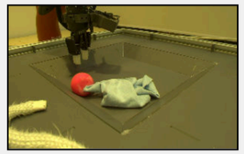
It's Homebase.

What is its web url?

It's homebase.co.uk.

What are they doing?

They are a home improvement retailer.



What do you think the robot is doing here?

The robot is picking up a ball.

What color is the ball?

It's red.

I want the ball to be hidden. Is there something the robot can use for that?

The robot can use a cloth to hide the ball.

What color is the cloth?

The cloth is blue.

This is a picture of two teddy bears on the moon.

What are they doing?

They are having a conversation.

What object are they using?

It looks like a computer.

Is this surprising?

Yes, it is surprising.

Why is this picture surprising to you?

I think it is surprising because teddy bears are not usually found on the moon.

What is the common thing about these three images?

They are all flamingos.

What is the difference between these three images?

The first one is a cartoon, the second one is a real flamingo, and the third one is a 3D model of a flamingo.

This is an apple with a sticker on it.

What does the sticker say?

The sticker says "iPod".

Where is the photo taken?

It looks like it's taken in a backyard.

Do you think it is printed or handwritten?

It looks like it's handwritten.

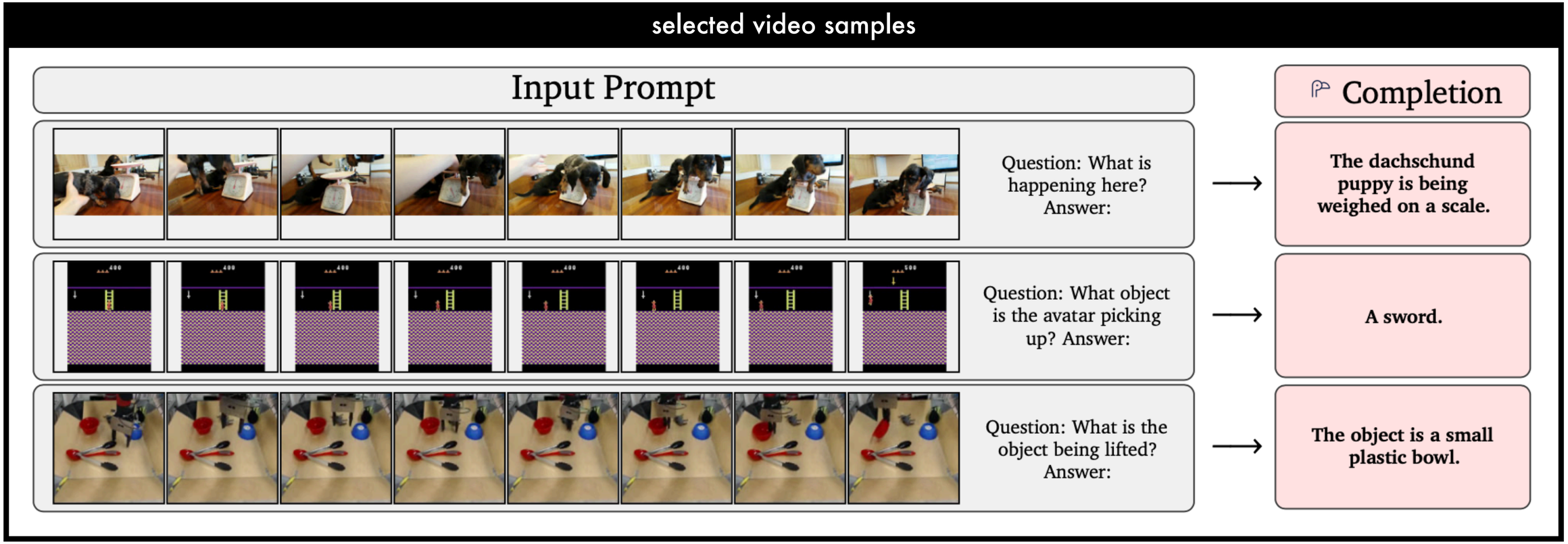
What color is the sticker?

It's white.

References/Image credits

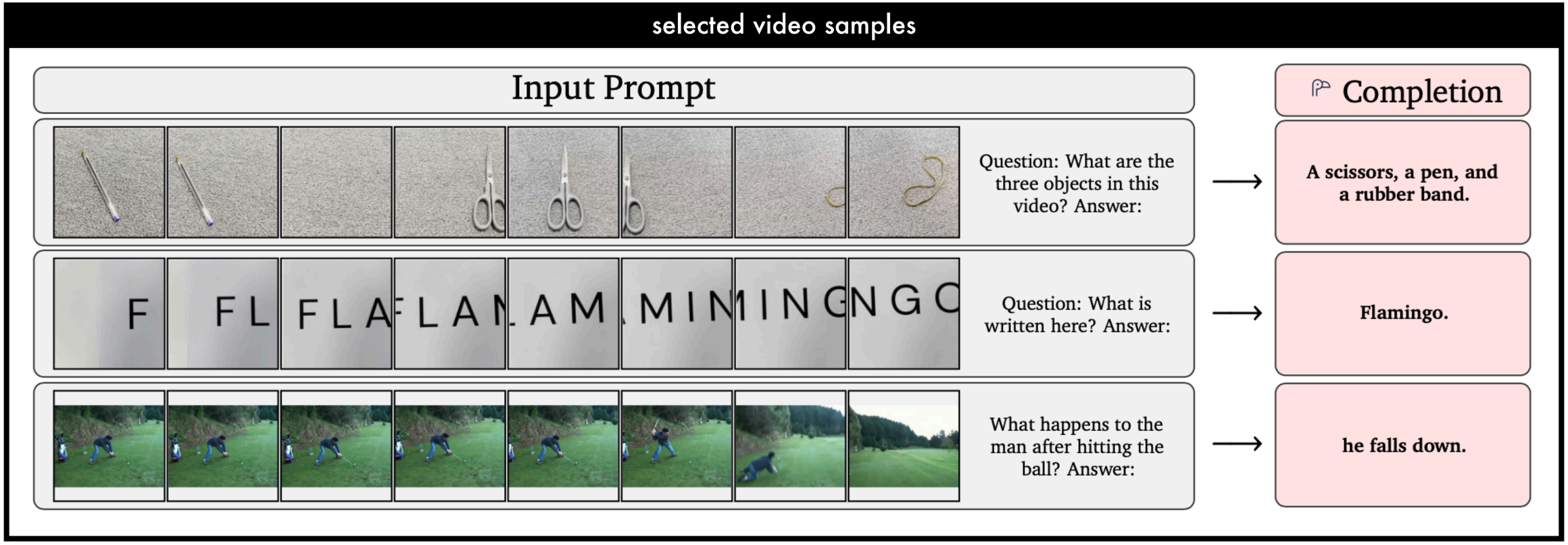
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Qualitative results - video



References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Qualitative results - more videos



References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Outline

- Motivation and challenges for multimodal generative modelling
- Related Work
- Flamingo model
- Evaluation datasets
- Comparison to state-of-the-art
- Ablations
- Qualitative Examples
- Discussion (limitations, trade-offs, opportunities, benefits, risks)

Flamingo limitations

Classification

Flamingo models lag contrastive models for classification

Possibly contrastive training optimises for retrieval

classification can be viewed as special case of retrieval (CLIP)

By contrast, language models do have this objective alignment

The work of Zhao et al. showed that language models are sensitive to prompt sample selection and their ordering

It is possible to calibrate to minimise these effects, but this requires assumptions on the label space (restrictive)

Future work: bridging the performance gap

Legacies of language models

Flamingo builds on pretrained language models, inheriting their weaknesses

Causal modelling is strictly less expressive than bidirectional modelling

Recent work by Wang et al. (2022) suggests non-causal masked language modelling with multi-task fine-tuning may be a better strategy for zero-shot generalisation

Challenge: if the expected output text is long, it is difficult to leverage enough shots

E.g. for VisDial, 1 shot is an image with 21 sentences, so 32 shots = 672 sentences

Results in 4096+ tokens - longer than the max training sequence length (2048)

This may explain why performance drops with more shots (16 vs 32) on VisDial

Language modelling suffers from poor sample efficiency (Brown et al., 2020)

Language model priors may also cause hallucinations and ungrounded guesses.

References

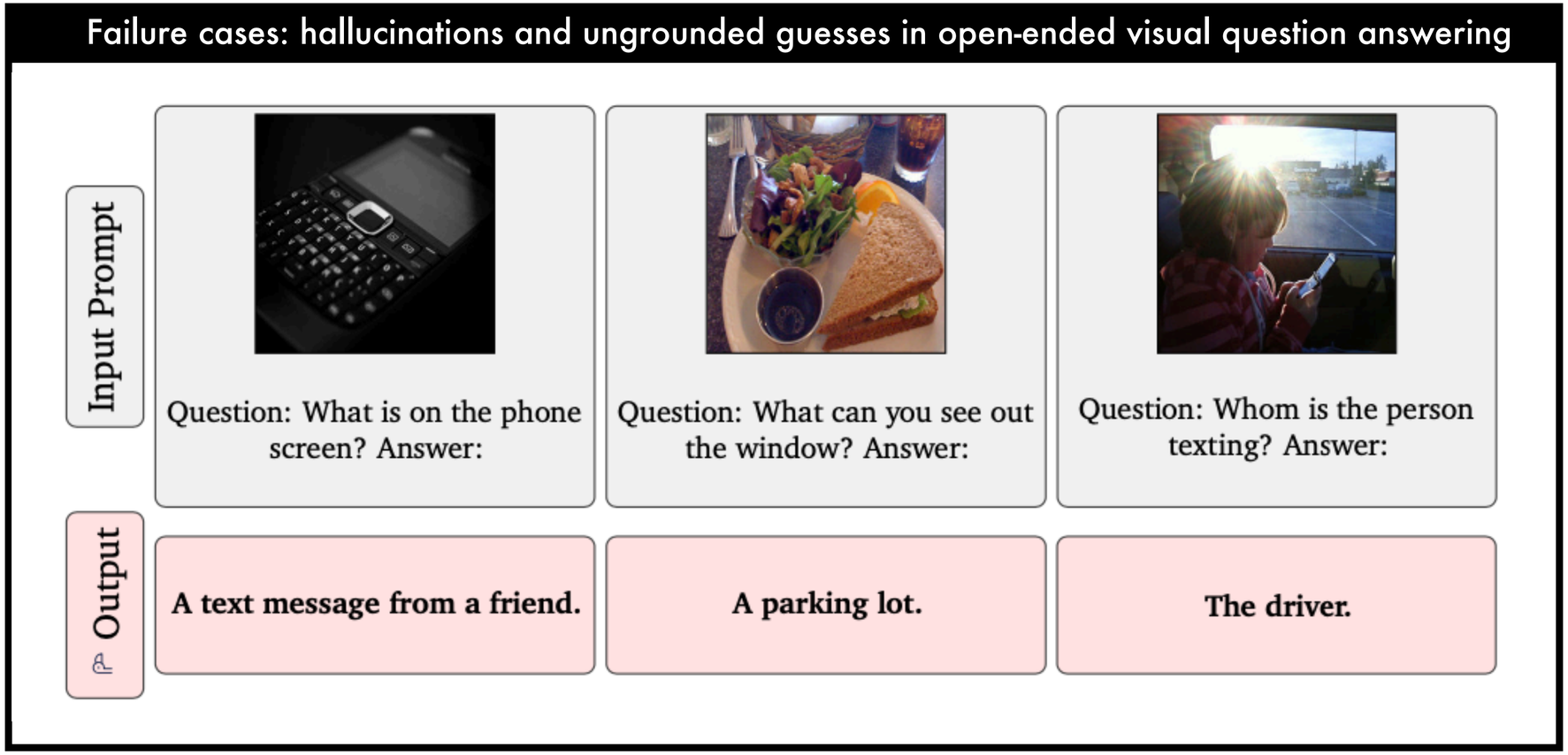
(CLIP) A. Radford et al., "Learning transferable visual models from natural language supervision", ICML (2021)

Z. Zhao et al., "Calibrate before use: Improving few-shot performance of language models", ICML (2021)

T. Wang et al., "What Language Model Architecture and Pretraining Objective Work Best for Zero-Shot Generalization?", arxiv (2022)

T. Brown et al., "Language models are few-shot learners", NeurIPS (2020)

Flamingo failures: hallucinations/ungrounded guesses



References/Image credits
J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

Trade-offs of few-shot learning methods

In-context learning: advantages

In-context learning has many advantages over fine-tuning:

- requires (almost) no hyperparameter tuning
- works reasonably in low-data regime (dozens of examples)
- only requires inference (simpler deployment)

By contrast, fine-tuning methods require:

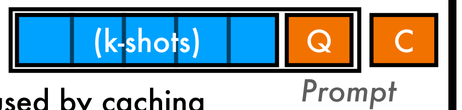
- carefully tuned design choices (learning rates, architecture)
- more data (e.g. thousands of examples) to work well

Advantages motivate choice of in-context learning for Flamingo

Takeaway

No "golden" few-shot method: best choice depends on the scenario

In-context learning: disadvantages



Inference compute cost: in-context learning cost scales

- linearly with the number of shots if the few-shot prompt can be re-used by caching
- quadratically with the number of shots if no such caching is possible

By contrast: gradient-based few-shot learning has constant complexity w.r.t. shots

Prompt sensitivity: in-context learning is sensitive to prompt order and format (Zhao et al., 2021)

Leveraging more shots: In-context learning performance plateaus as shots increase (e.g. >32)

By contrast: gradient-based approaches tend to continue to benefit from more examples

RICES helps to some extent, but there are still issues for larger numbers of examples per class

Task location: several works (Min et al., 2022; Reynolds and McDonell, 2021) suggest in-context learning may not be learning a task, but instead identifying the task to be performed

A few examples may therefore help task location, but its possible the model can't do anything more than task location from them, and hence cannot scale up to usefully use more examples

References

Z. Zhao et al., "Calibrate before use: Improving few-shot performance of language models", ICML (2021)

S. Min et al., "Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?", arxiv (2022)

L. Reynolds and K. McDonell, "Prompt programming for large language models: Beyond the few-shot paradigm", CHI (2021)

Flamingo opportunities

Extending the visual and text interface

Natural language provides a versatile interface to:

- provide descriptions of visual tasks
- generate model outputs
- estimate conditional likelihoods over candidate outputs

However, it is cumbersome for structured prediction, ill suited for:

- conditioning on/predicting structures like bounding boxes
- dense predictions (over space or time)
- continuous predictions (like optical flow)

Further modalities (like audio) could extend the interface

Scaling laws for vision-language models

Flamingo scales up to 80B parameters and provides some insights about scaling behaviour

Scaling laws were studied for language (Kaplan et al., 2020) and vision (Zhai et al., 2021)

There is limited work understanding scaling for vision-language models

Rather than focusing on perplexity, downstream task performance may be a better metric

References/Image credits
J. Kaplan et al., "Scaling laws for neural language models", arxiv (2020)
X. Zhai et al., "Scaling vision transformers", arxiv (2021)

Flamingo benefits

Accessibility

Flamingo can be trained with minimal examples and used through a chat-like interface for open-ended dialogue

This could enable non-expert users to apply Flamingo in low-resource settings

Example: Flamingo works well on VizWiz

Dialogue interface could help highlight issues with bias/toxicity

Model recycling

Although costly to train, Flamingo demonstrates how to leverage frozen pretrained vision encoders and language models

This suggests new modalities can be introduced into frozen models

Could help with reducing environmental impact (Strubell et al., 2019)

References
E. Strubell et al., "Energy and policy considerations for deep learning in NLP", ACL (2019)

Flamingo risks and mitigation strategies

Inherited risks of large language models

Flamingo relies heavily on a pretrained language model

With no input images, it defaults to language model behaviour

Result: offensive language, stereotypes, private info leakage

Flamingo is based on Chinchilla (Hoffmann et al., 2020)

slightly less gendered biased than prior models, but still biased

Chinchilla also has relatively low toxicity, as measured using the PerspectiveAPI toxicity score on 25,000 samples

Potential mitigations: social/public policy interventions (regulation and guidelines), research on AI Ethics/NLP, better benchmarks

Toxicity when prompted with images

Some Flamingo captions were tagged as toxic by PerspectiveAPI

However, on manual inspection, no clear toxicity was found

Toxic outputs not observed for with "safe-for-work" imagery

Gender and racial biases

Conduct a study to assess captioning bias following Zhao et al. (2021)

Evaluate how performance varies on COCO as a function of gender and skin colour

	CIDEr difference		CIDER
	female - male = Δ	darker - lighter = Δ	overall
Flamingo, 0 shot	0.899 – 0.870 = +0.029 (p = 0.52)	0.955 – 0.864 = +0.091 (p = 0.25)	0.843
Flamingo, 32 shots	1.172 – 1.142 = +0.030 (p = 0.54)	1.128 – 1.152 = –0.025 (p = 0.76)	1.138

No statistically significant differences were observed.

Flamingo for mitigation

Flamingo could be used for filtering purposes for toxic samples in the training data

During evaluation, models adapted on filtered data could be used to down-rank/ exclude outputs that do not meet desired standards

Flamingo performance on HatefulMemes suggesting it may be well-suited for this task

Could be used for "red-teaming" to identify issues in other models (Perez et al., 2022)

As shown in the qualitative examples, Flamingo can, in cases, explain its own outputs

References/Image credits

(Chinchilla) J. Hoffmann et al., "Training Compute-Optimal Large Language Models", arxiv (2022)

D. Zhao et al., "Understanding and Evaluating Racial Biases in Image Captioning", *ICCV* (2021)

J-B. Alayrac et al., "Flamingo: a Visual Language Model for Few-Shot Learning", arxiv (2022)

E. Perez et al., "Red teaming language models with language models", arxiv (2022)

Summary

Summary

Flamingo is a "general-purpose" family of models that can be applied to images and video data with limited training data

State-of-the-art results a variety of tasks

Qualitative examples demonstrating interactive abilities

In summary, a highly flexible vision and language model