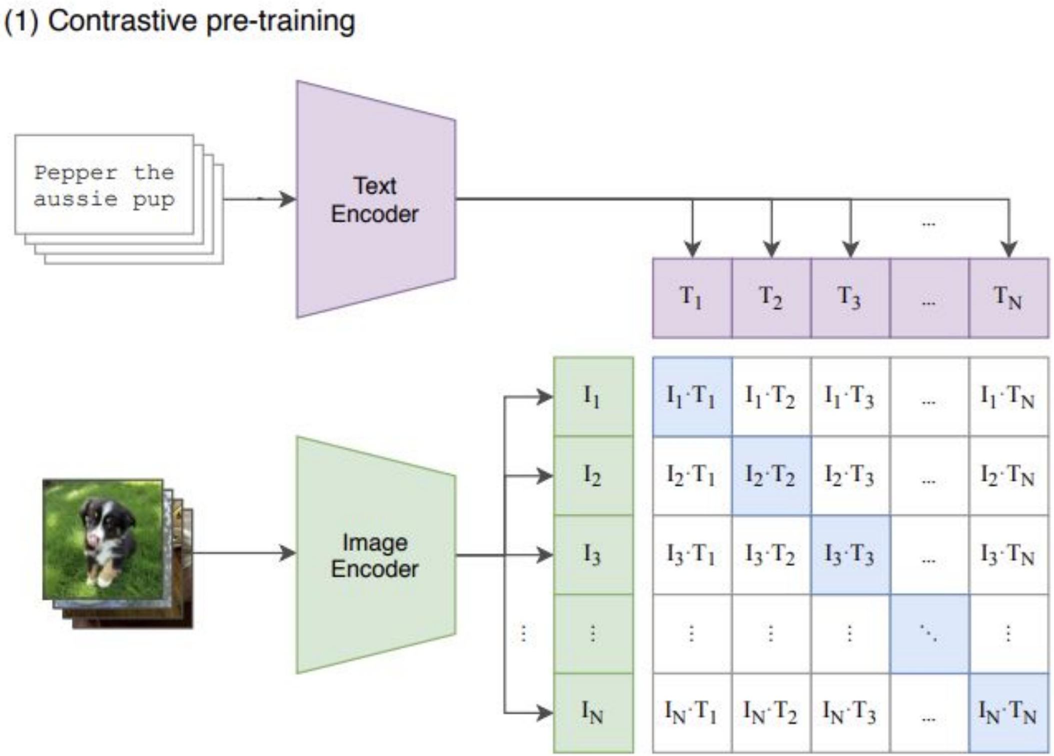
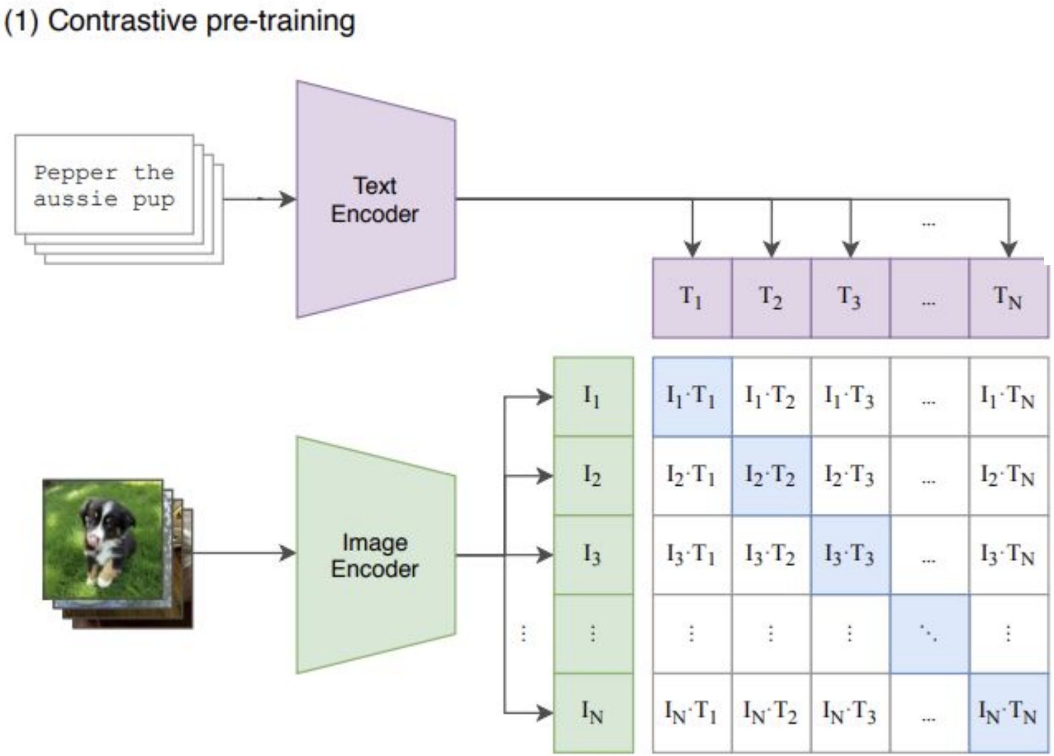
Visual Encoders

Vision+Language Seminar (Fall 2024)

CLIP: Contrastive Language-Image Pretraining (Radford et al, 2021)



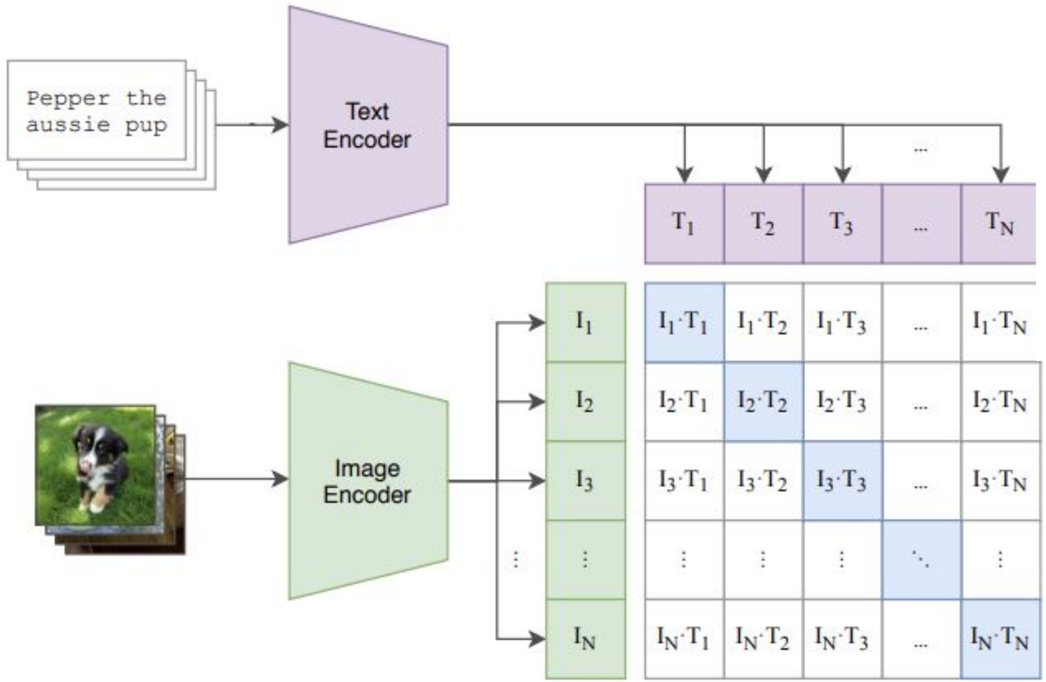
CLIP: Contrastive Language-Image Pretraining (Radford et al, 2021)



$$I : B \times H \times W \times 3$$
$$T : B \times S$$
$$e_I = \text{ImageEncoder}(I) \ \# \ (\text{B, D})$$
$$e_T = \text{TextEncoder}(T) \ \# \ (\text{B, D})$$

CLIP: Contrastive Language-Image Pretraining (Radford et al, 2021)

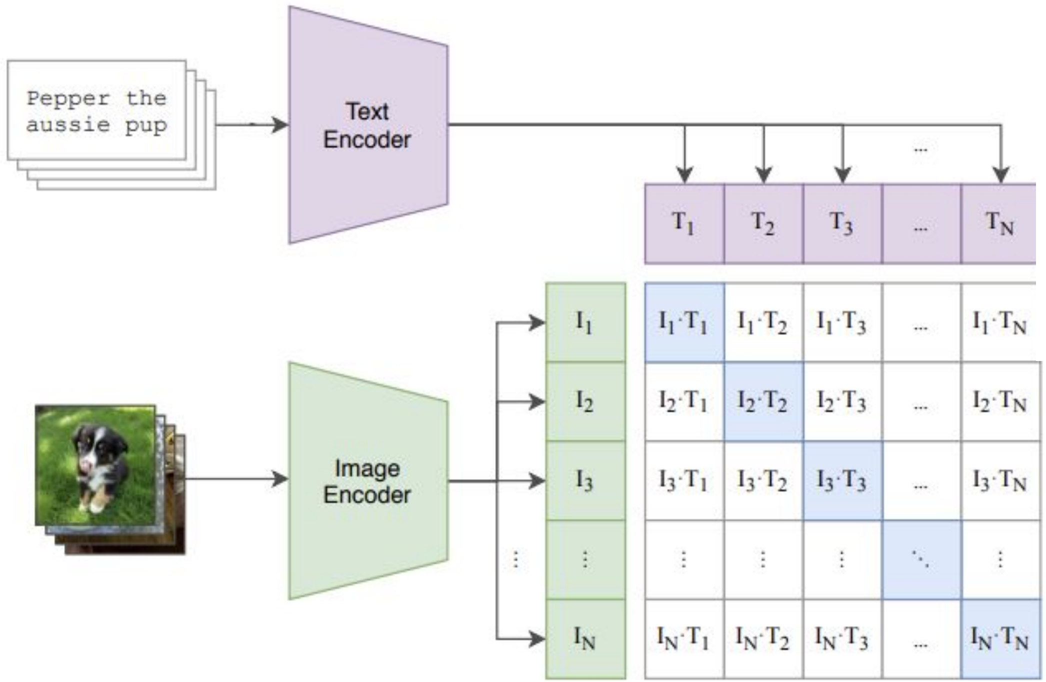
(1) Contrastive pre-training



$$I : B \times H \times W \times 3$$
$$T : B \times S$$
$$e_I = \text{ImageEncoder}(I) \ \# \ (\text{B, D})$$
$$e_T = \text{TextEncoder}(T) \ \# \ (\text{B, D})$$
$$e_I = \text{L2\_norm}(e_I)$$
$$e_T = \text{L2\_norm}(I_T)$$
$$cosine\_sim = e_I e_T^T \exp(\tau) \ \# \ (\text{B, B})$$

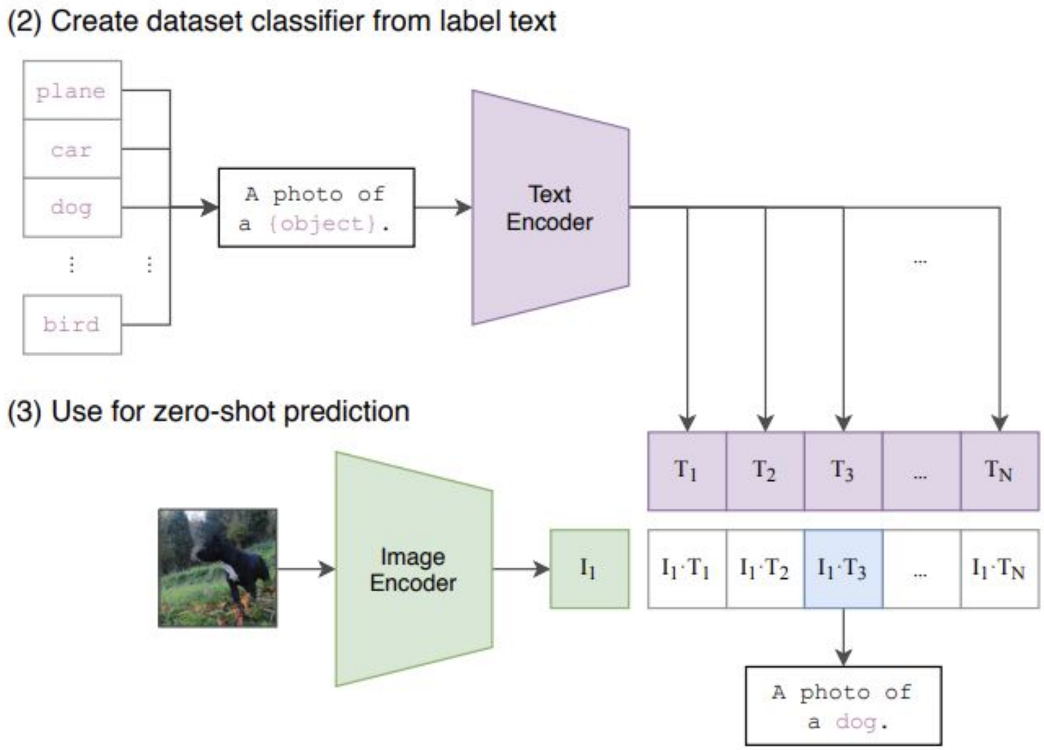
CLIP: Contrastive Language-Image Pretraining (Radford et al, 2021)

(1) Contrastive pre-training

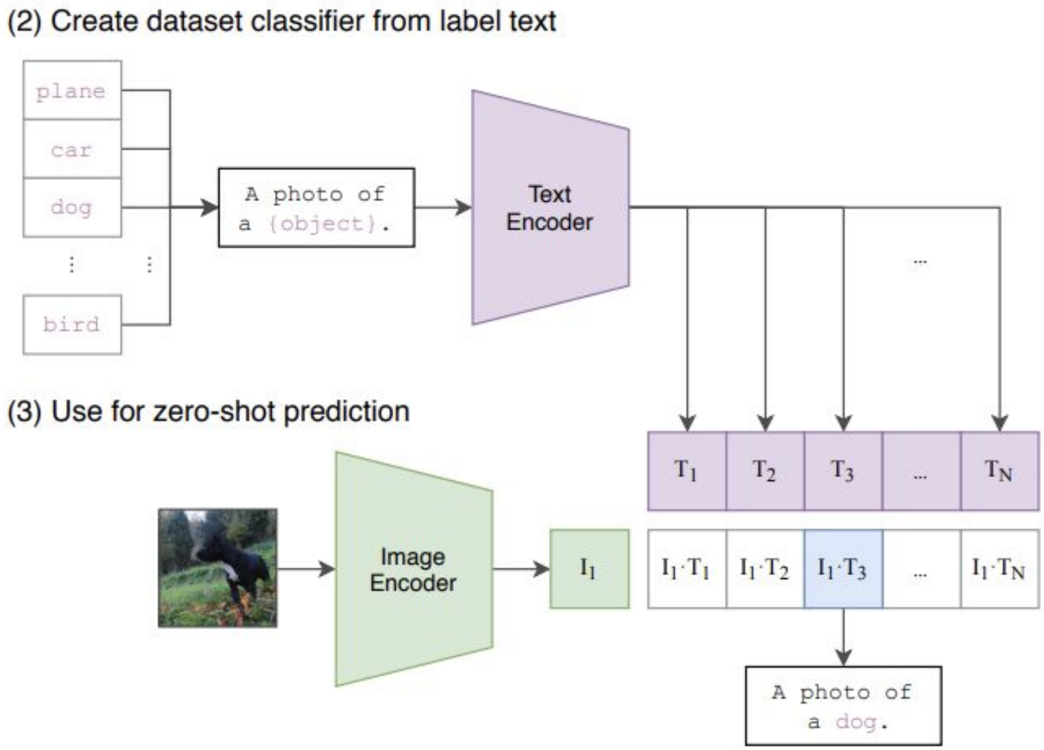


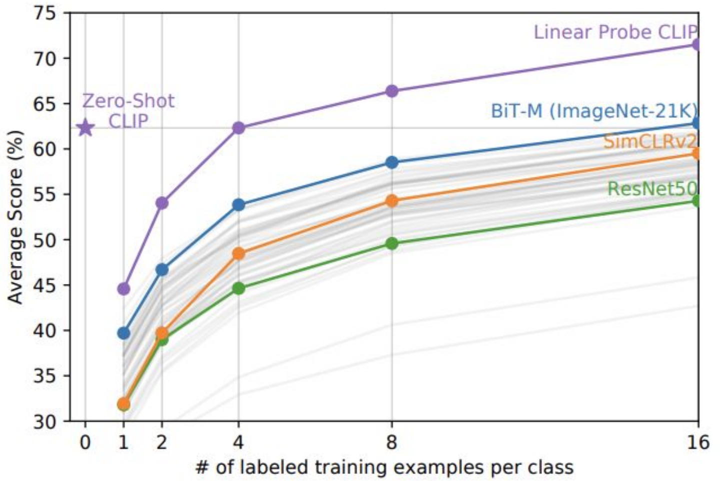
$$I : B \times H \times W \times 3$$
$$T : B \times S$$
$$e_I = \text{ImageEncoder}(I) \ \# \ (\text{B, D})$$
$$e_T = \text{TextEncoder}(T) \ \# \ (\text{B, D})$$
$$e_I = \text{L2\_norm}(e_I)$$
$$e_T = \text{L2\_norm}(I_T)$$
$$cosine\_sim = e_I e_T^T \exp(\tau) \ \# \ (\text{B, B})$$
$$logits1 = \text{log\_softmax}(cosine\_sim, \text{dim} = 0)$$
$$logits2 = \text{log\_softmax}(cosine\_sim, \text{dim} = 1)$$
$$loss1 = (logits1 * Identity(B)).sum()$$
$$loss2 = (logits2 * Identity(B)).sum()$$
$$loss = (loss1 + loss2)/2$$

CLIP: Contrastive Language-Image Pretraining (Radford et al, 2021)

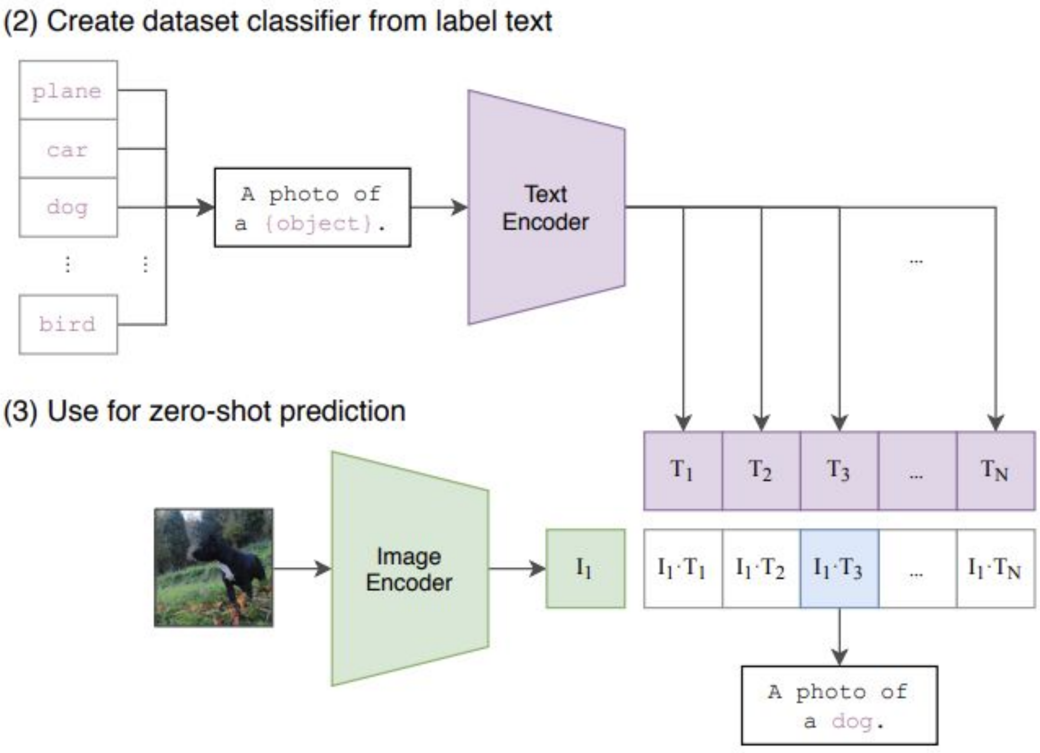


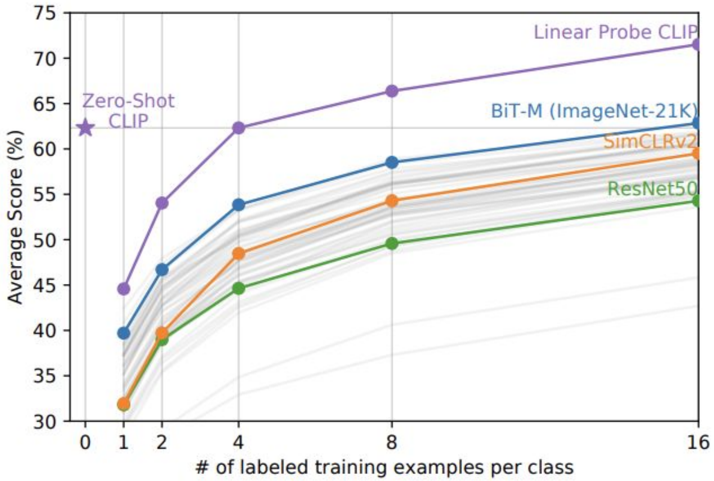
CLIP: Contrastive Language-Image Pretraining (Radford et al, 2021)





CLIP: Contrastive Language-Image Pretraining (Radford et al, 2021)

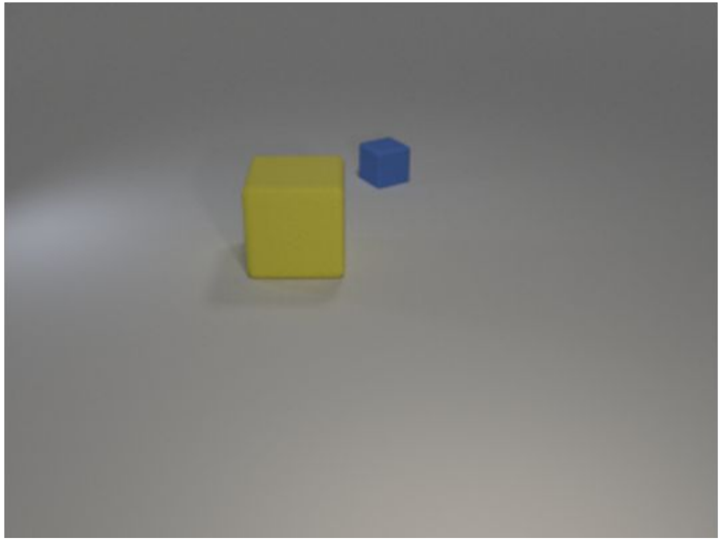




certain search volume. Finally all WordNet (Miller, 1995) synsets not already in the query list are added.

We approximately class balance the results by including up to 20,000 (image, text) pairs per query. The resulting

Can we use CLIP to resolve spatial relations?



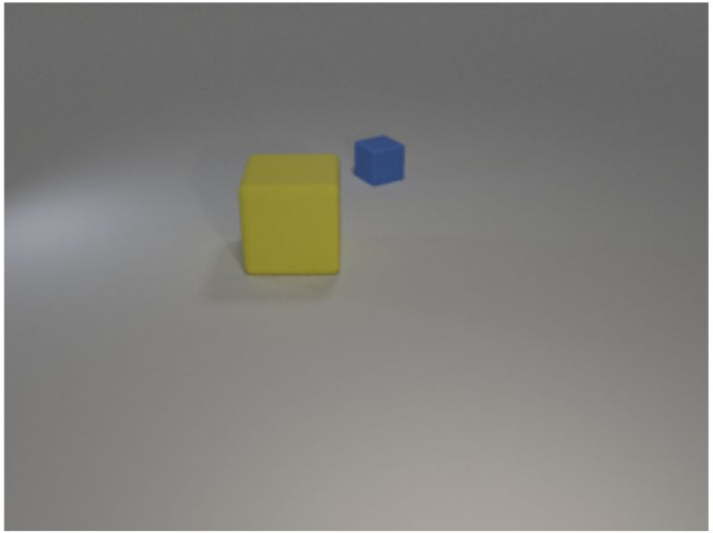
“a yellow cube is in front of a blue cube”

`“a yellow cube is behind a blue cube”`

Subramanian et al, 2022

Can we use CLIP to resolve spatial relations?

Control task



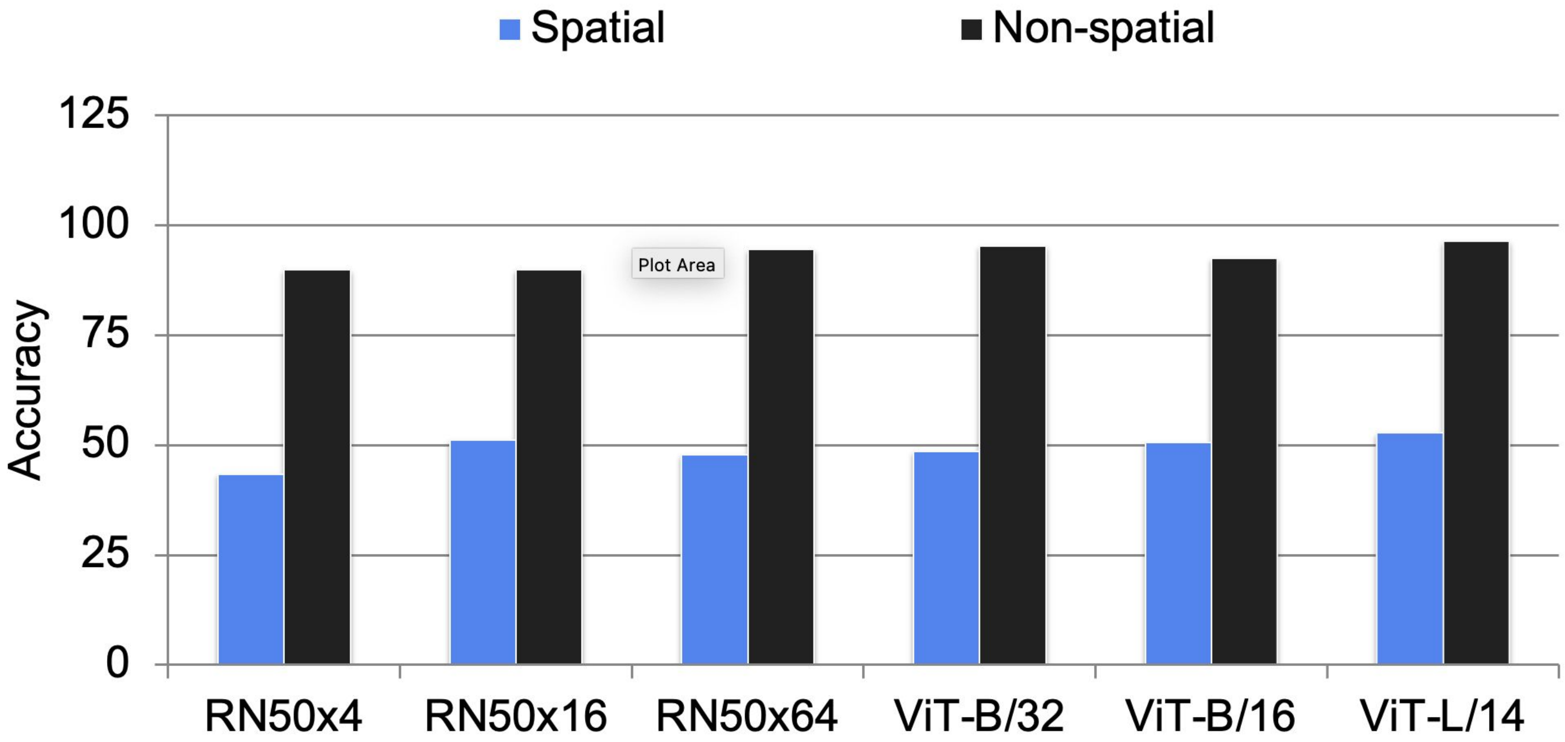
“a blue cube and a yellow cube”

“a blue cube and a yellow sphere”

→ CLIP

Subramanian et al, 2022

Can we use CLIP to resolve spatial relations?



Subramanian et al, 2022

SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid

$$-\frac{1}{2|\mathcal{B}|}\sum_{i=1}^{|\mathcal{B}|}\left(\overbrace{\log\frac{e^{t\mathbf{x}_i\cdot\mathbf{y}_i}}{\sum_{j=1}^{|\mathcal{B}|}e^{t\mathbf{x}_i\cdot\mathbf{y}_j}}}^{\text{image}\rightarrow\text{text softmax}}+\overbrace{\log\frac{e^{t\mathbf{x}_i\cdot\mathbf{y}_i}}{\sum_{j=1}^{|\mathcal{B}|}e^{t\mathbf{x}_j\cdot\mathbf{y}_i}}}^{\text{text}\rightarrow\text{image softmax}}\right)$$

$$\downarrow$$

$$-\frac{1}{|\mathcal{B}|}\sum_{i=1}^{|\mathcal{B}|}\sum_{j=1}^{|\mathcal{B}|}\underbrace{\log\frac{1}{1+e^{z_{ij}(-t\mathbf{x}_i\cdot\mathbf{y}_j+b)}}}_{\mathcal{L}_{ij}}$$

SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid

$$-\frac{1}{2|\mathcal{B}|}\sum_{i=1}^{|\mathcal{B}|}\left(\overbrace{\log\frac{e^{t\mathbf{x}_i\cdot\mathbf{y}_i}}{\sum_{j=1}^{|\mathcal{B}|}e^{t\mathbf{x}_i\cdot\mathbf{y}_j}}}^{\text{image}\rightarrow\text{text softmax}}+\overbrace{\log\frac{e^{t\mathbf{x}_i\cdot\mathbf{y}_i}}{\sum_{j=1}^{|\mathcal{B}|}e^{t\mathbf{x}_j\cdot\mathbf{y}_i}}}^{\text{text}\rightarrow\text{image softmax}}\right)$$

$$\downarrow$$

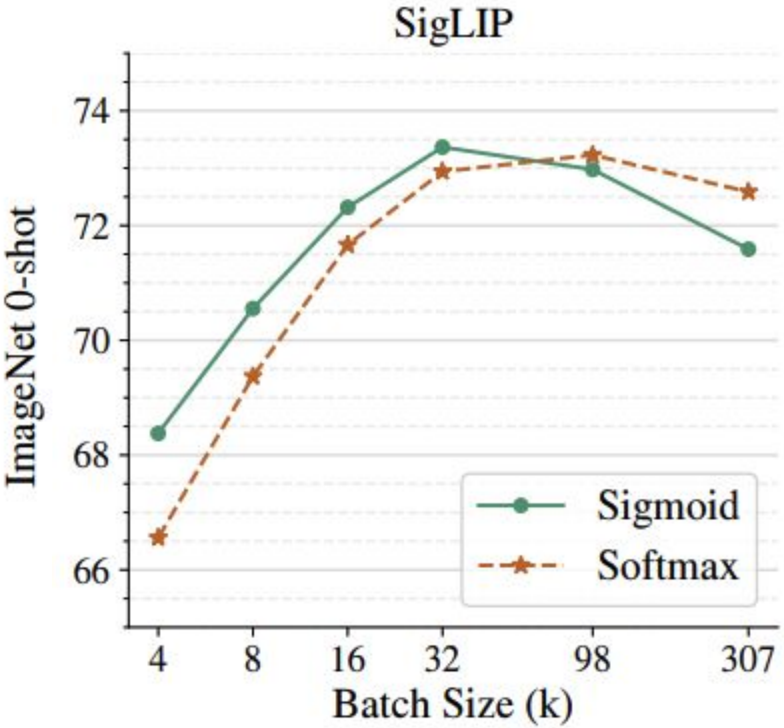
$$-\frac{1}{|\mathcal{B}|}\sum_{i=1}^{|\mathcal{B}|}\sum_{j=1}^{|\mathcal{B}|}\underbrace{\log\frac{1}{1+e^{z_{ij}(-t\mathbf{x}_i\cdot\mathbf{y}_j+b)}}}_{\mathcal{L}_{ij}}$$

Publicly available models

Method	Image Encoder		ImageNet-1k				COCO R@1	
	ViT size	# Patches	Validation	v2	ReaL	ObjectNet	I → T	T → I
CLIP	L	256	75.5	69.0	-	69.9	56.3	36.5
SigLIP	L	256	**80.5**	**74.2**	**85.9**	**77.9**	**69.5**	**51.1**
CLIP	L	576	76.6	72.0	-	70.9	57.9	37.1
SigLIP	L	576	**82.1**	**75.9**	**87.0**	**81.0**	**70.6**	**52.7**

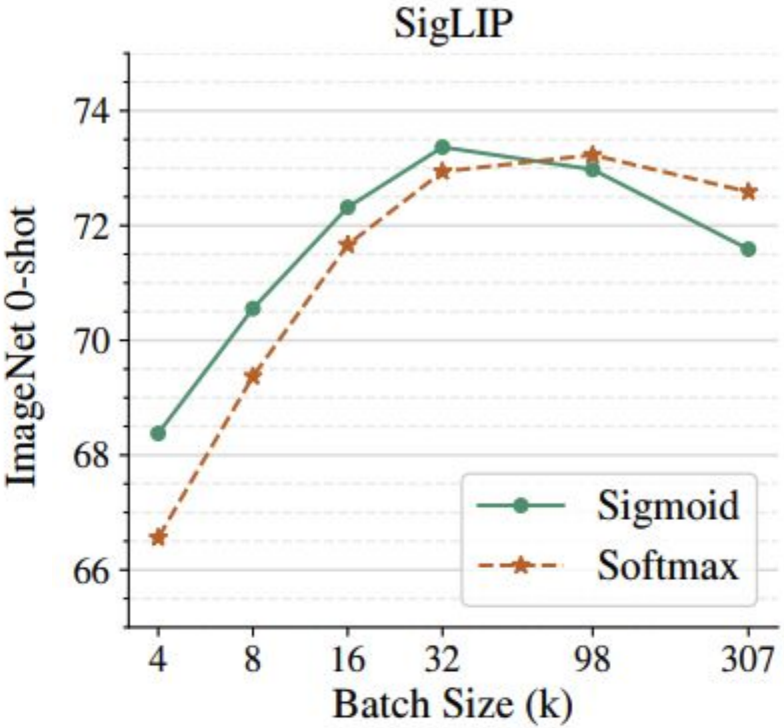
SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid

Comparison with Same Training Data



SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid

Comparison with Same Training Data



3 observations:

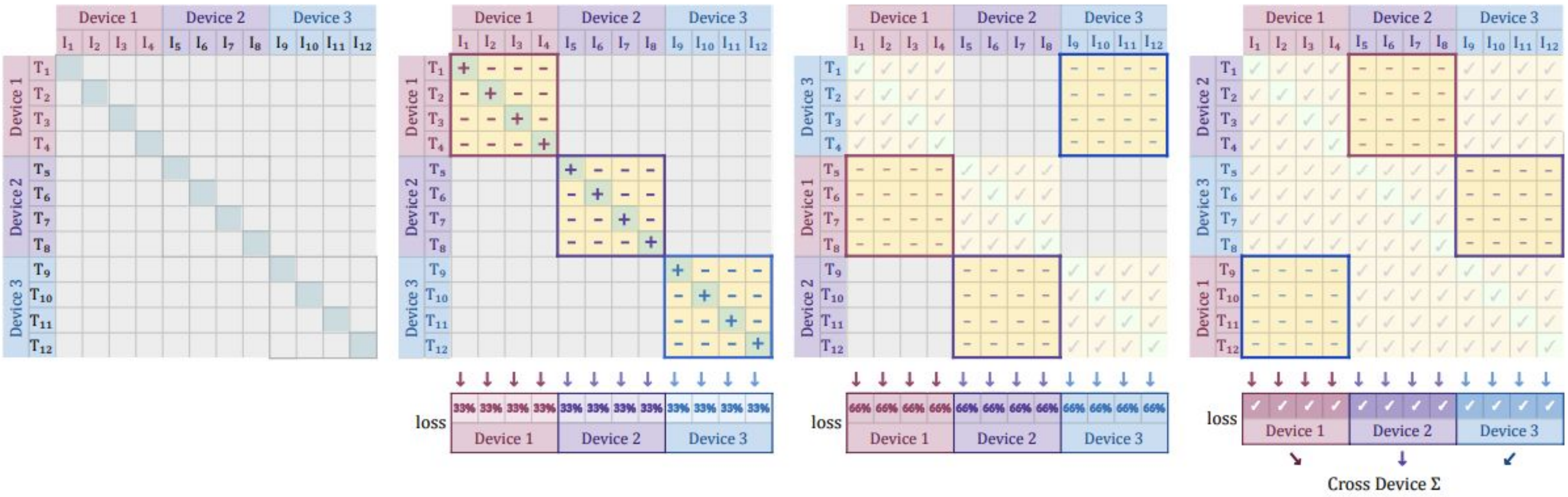
1. Sigmoid vs. Softmax difference is mainly visible at small BS
2. Sigmoid worse than Softmax at largest BS
3. Both methods worse at largest BS than medium BS

SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid

Contrastive training typically utilizes data parallelism. Computing the loss when data is split across D devices necessitates gathering all embeddings [59] with expensive all-gathers and, more importantly, the materialization of a memory-intensive $|\mathcal{B}| \times |\mathcal{B}|$ matrix of pairwise similarities.

days. We also present from-scratch results in the bottom rows of Table 1: with 32 TPUv4 chips for only two days, SigLIP achieves 72.1% 0-shot accuracy. This presents a significant training cost reduction e.g. compared to CLIP (approx. 2500 TPUv3-days for 72.6%) reported in [30].

SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid



SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid

Should we prune negative pairs? (ratio is 16K:1 by default) If so, how?

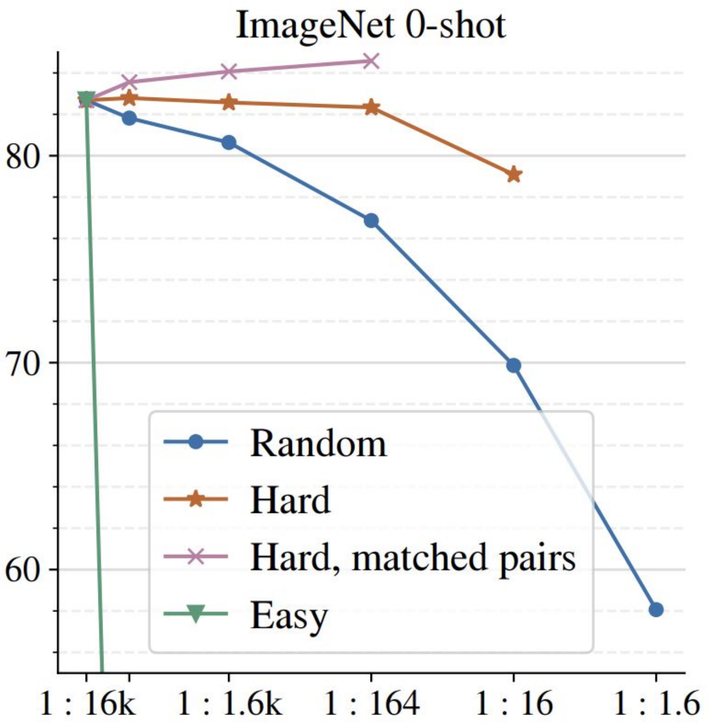
SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid

Should we prune negative pairs? (ratio is 16K:1 by default) If so, how?

- **Random:** Randomly choose negative pairs to mask.
- **Hard:** Keep hardest negative pairs (highest loss).
- **Easy:** Keep easiest negatives pairs (lowest loss).
- **Hard + matching total pairs seen:** Masking examples while training for a fixed number of steps does decrease the total number of *pairs* seen during training. Hence in the *matched pairs* setting, we increase the number of training steps by the masking ratio in order to keep the number of pairs seen constant.

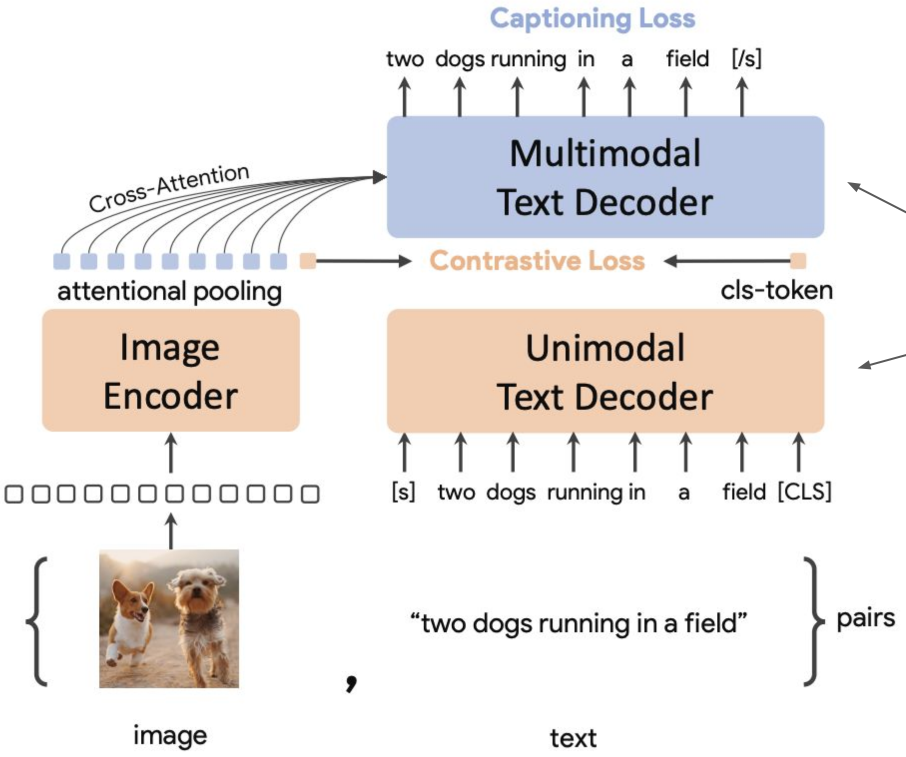
SigLIP (Zhai et al, 2023): Replace Softmax with Sigmoid

Should we prune negative pairs? (ratio is 16K:1 by default) If so, how?



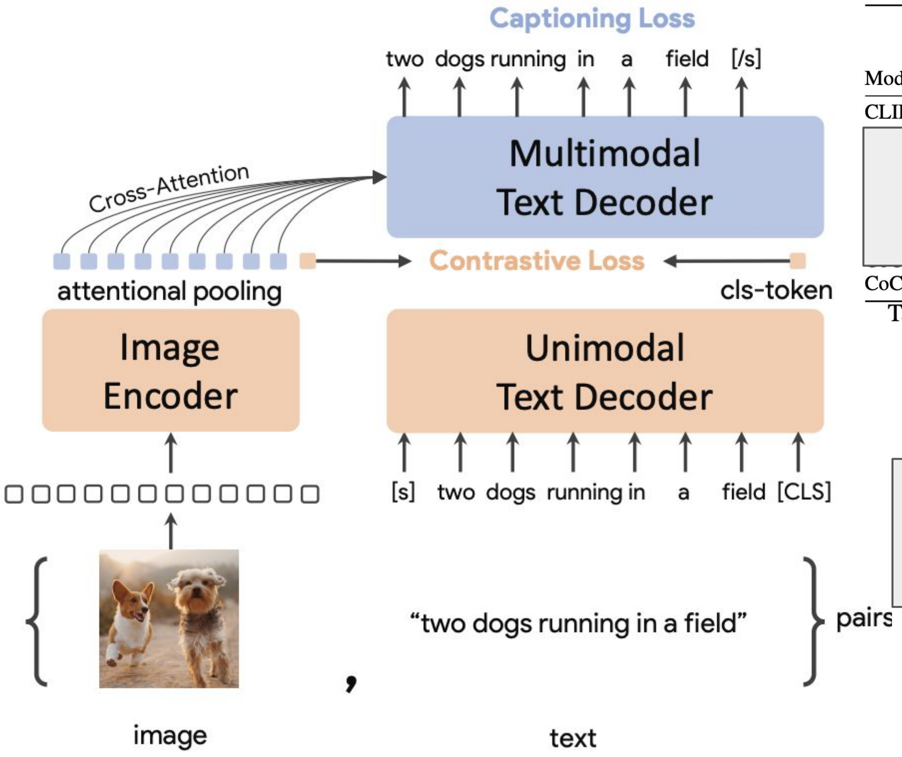
- **Random:** Randomly choose negative pairs to mask.
- **Hard:** Keep hardest negative pairs (highest loss).
- **Easy:** Keep easiest negatives pairs (lowest loss).
- **Hard + matching total pairs seen:** Masking examples while training for a fixed number of steps does decrease the total number of *pairs* seen during training. Hence in the *matched pairs* setting, we increase the number of training steps by the masking ratio in order to keep the number of pairs seen constant.

CoCa: Contrastive Captioners are Image-Text Foundation Models (Yu et al, 2022)



Top n layers and bottom n layers of a single decoder transformer

CoCa: Contrastive Captioners are Image-Text Foundation Models (Yu et al, 2022)



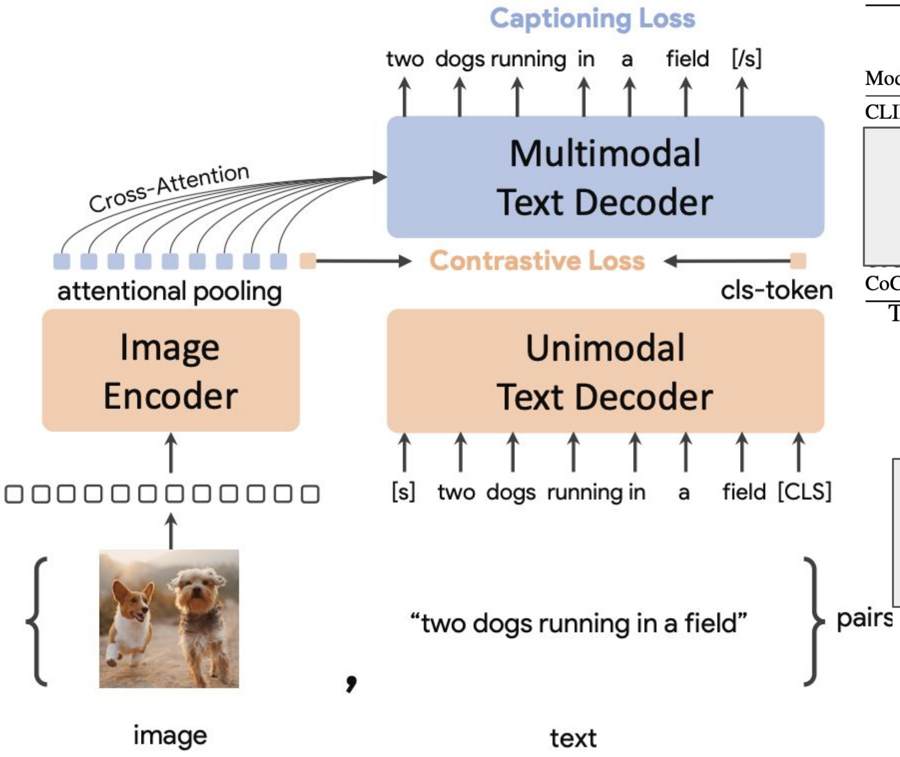
	Flickr30K (1K test set)						MSCOCO (5K test set)					
	Image → Text			Text → Image			Image → Text			Text → Image		
Model	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP [12]	88.0	98.7	99.4	68.7	90.6	95.2	58.4	81.5	88.1	37.8	62.4	72.2
CoCa	**92.5**	**99.5**	**99.9**	**80.4**	**95.7**	**97.7**	**66.3**	**86.2**	**91.8**	**51.2**	**74.2**	**82.0**

Table 3: Zero-shot image-text retrieval results on Flickr30K [62] and MSCOCO [63] datasets.

Model	ImageNet	ImageNet-A	ImageNet-R	ImageNet-V2	ImageNet-Sketch	ObjectNet	Average
CLIP [12]	76.2	77.2	88.9	70.1	60.2	72.3	74.3
CoCa	**86.3**	**90.2**	**96.5**	**80.7**	**77.6**	**82.7**	**85.7**

Table 4: Zero-shot image classification results on ImageNet [9], ImageNet-A [64], ImageNet-R [65], ImageNet-V2 [66], ImageNet-Sketch [67] and ObjectNet [68].

CoCa: Contrastive Captioners are Image-Text Foundation Models (Yu et al, 2022)



	Flickr30K (1K test set)						MSCOCO (5K test set)					
	Image → Text			Text → Image			Image → Text			Text → Image		
Model	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
CLIP [12]	88.0	98.7	99.4	68.7	90.6	95.2	58.4	81.5	88.1	37.8	62.4	72.2
CoCa	**92.5**	**99.5**	**99.9**	**80.4**	**95.7**	**97.7**	**66.3**	**86.2**	**91.8**	**51.2**	**74.2**	**82.0**

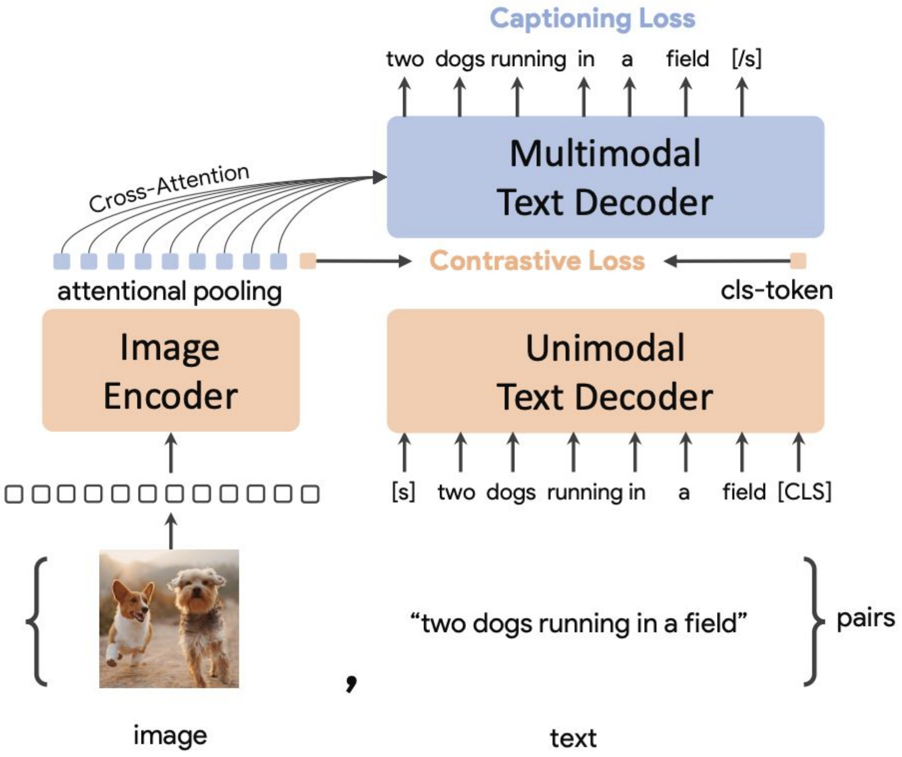
Table 3: Zero-shot image-text retrieval results on Flickr30K [62] and MSCOCO [63] datasets.

Model	ImageNet	ImageNet-A	ImageNet-R	ImageNet-V2	ImageNet-Sketch	ObjectNet	Average
CLIP [12]	76.2	77.2	88.9	70.1	60.2	72.3	74.3
CoCa	**86.3**	**90.2**	**96.5**	**80.7**	**77.6**	**82.7**	**85.7**

Table 4: Zero-shot image classification results on ImageNet [9], ImageNet-A [64], ImageNet-R [65], ImageNet-V2 [66], ImageNet-Sketch [67] and ObjectNet [68].

scale alt-text data and annotated images by treating all labels simply as texts. We use the JFT-3B dataset [21] with label names as the paired texts, and the ALIGN dataset [13] with noisy alt-texts.

CoCa: Contrastive Captioners are Image-Text Foundation Models (Yu et al, 2022)



loss	ZS	VQA	TPU cost
$\mathcal{L}_{\text{Con}}$	70.7	59.2	1×
$\mathcal{L}_{\text{Cap}}$	-	68.9	1.17×
$\mathcal{L}_{\text{CoCa}}$	**71.6**	**69.0**	**1.18**×

(b) Training objectives ablation.

Discussion Questions (about SigLIP)

- Latest MLLMs usually use SigLIP instead of CLIP as the default vision encoder because people find SigLIP is consistently better on different MLLM benchmarks. However, we are still not clear is this advantage due to the loss design or the data?
- Prior work found that increasing batch size improves the training with contrastive loss objective. Why is the peak performance at 32k batch size? What's the intuition behind this result? Is it due to large class imbalance and that negative examples grow quadratically (N^2-N) compared to linear growth of positive examples (N)? How can we improve large-batch size performance (1M)?
- The results in Figure 7, which show that the proposed method leads to models which have improved noise robustness, suggests that the sigmoid loss is having some kind of regularization effect beyond the softmax loss. What could be the mechanism/explanation behind this?