BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation - Junnan Li, Dongxu Li, Caiming Xiong, Steven Hoi (Salesforce)

Presentation Author
Mathieu Ravaut

Nanyang Technological University

Table of content

Introduction

- **Problem** : Most models are either **encoder-based**, or **encoder-decoder** models. Encoder-based models are typically good at **understanding** tasks, while encoder-decoder are good at **generation** tasks. But **none** of the two structures excels at **both** types of tasks together.

Introduction

- **Problem** : Most models are either encoder-based, or encoder-decoder models. Encoder-based models are typically good at understanding tasks, while encoder-decoder are good at generation tasks. But none of the two structures excels at both types of tasks together.
- **Proposed solution** : **Multimodal mixture of Encoder-Decoder (MED)**, a new architecture which can serve both for unimodal encoding tasks, and decoding tasks.

Introduction

- **Problem** : Most vision-language pre-training approaches use image-text pairs **sourced from the web**. However, there is a **lot of noise**, and the text does not always accurately describe the image. This problem has not been properly addressed yet, as the authors nicely put it : *However, the negative impact of the noise has been largely overlooked, shadowed by the performance gain obtained from scaling up the dataset.*

Introduction

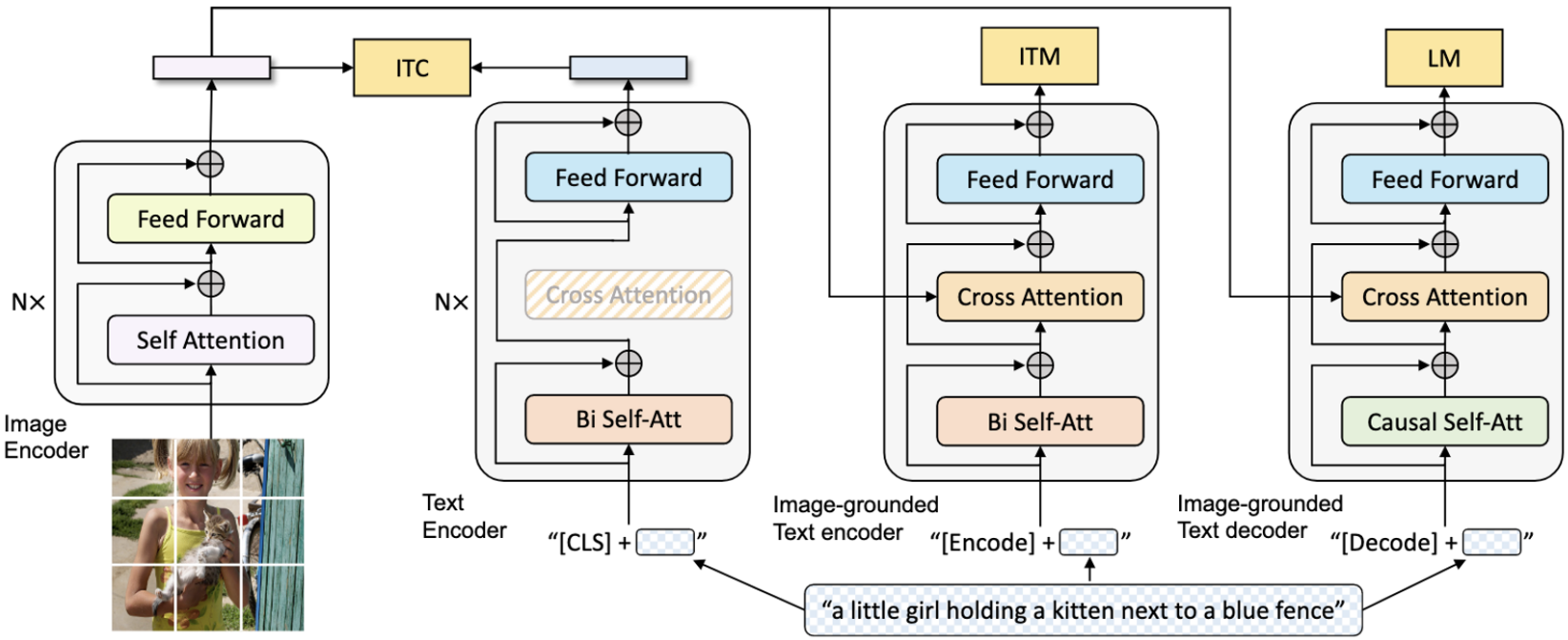
- **Problem** : Most vision-language pre-training approaches use image-text pairs sources from the web. However, there is a lot of noise, and the text does not always accurately describe the image. This problem has not been properly addressed yet, as the authors nicely put it : *However, the negative impact of the noise has been largely overlooked, shadowed by the performance gain obtained from scaling up the dataset.*
- **Proposed solution** : **Captioning and Filtering (CapFilt)**, a dataset bootstrapping method. It uses the proposed architecture elegantly to clean up image-text pre-training pairs.

Introduction

- BLIP offers to tackle both this model architecture limitation, and this data limitation, with a **new vision-language pre-training approach**.
- It achieves SOTA results on 7 (!) vision-language tasks :
  - Image-text retrieval
  - Image captioning
  - Visual question-answering (VQA)
  - Visual dialog
  - Natural language visual reasoning (NLVR)
  - Text-to-video retrieval
  - Video question-answering

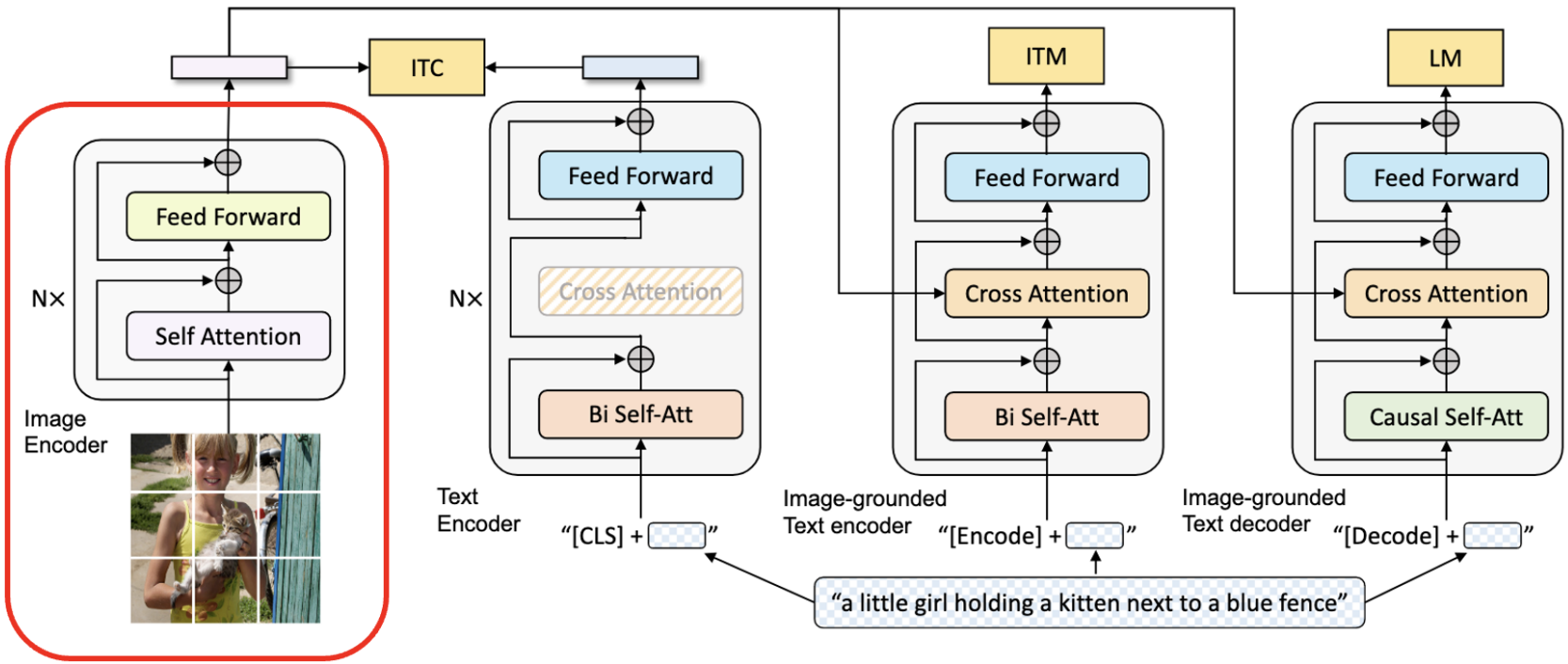
Approach : overview

BLIP uses **three encoders** : image, text, image-grounded text and **one** decoder : image-grounded text decoder



Approach : image encoder

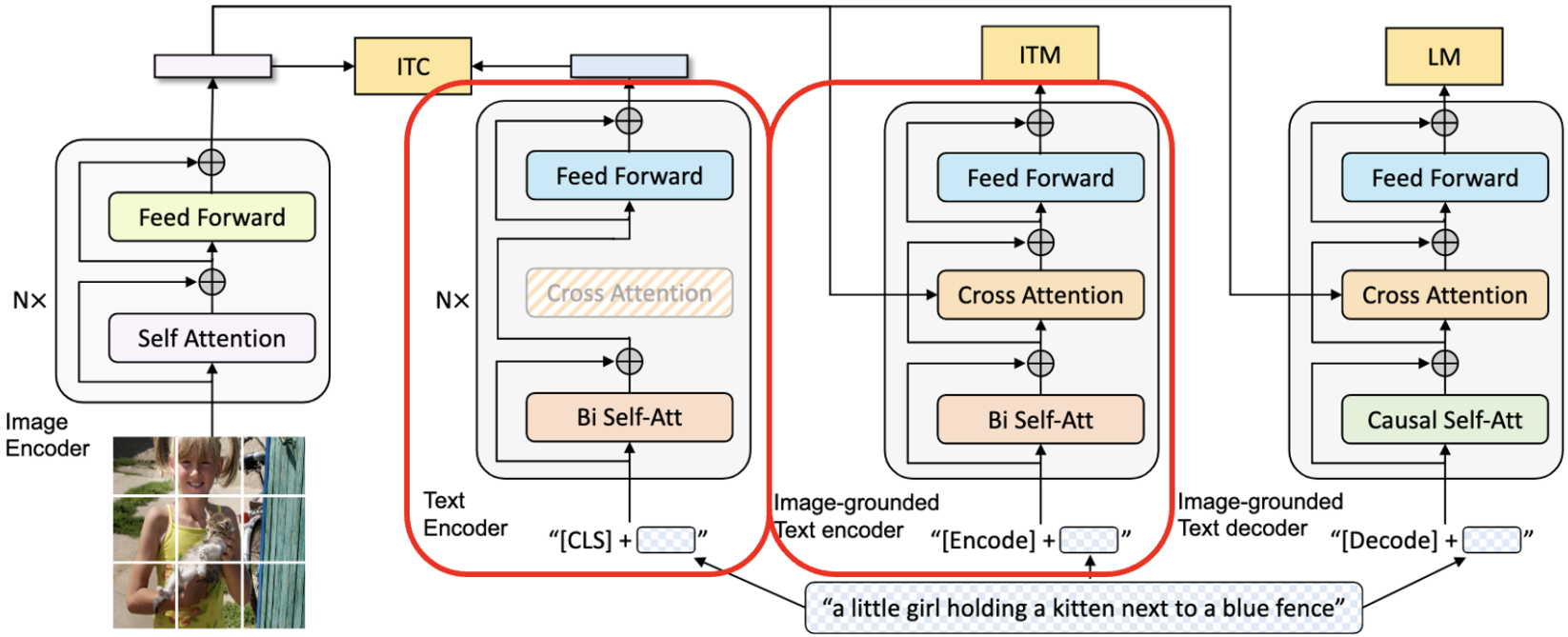
The vision encoder is a **Vision Transformer** (ViT) (pre-trained on ImageNet), discarding computationally-intensive object detectors.



Approach : text encoders

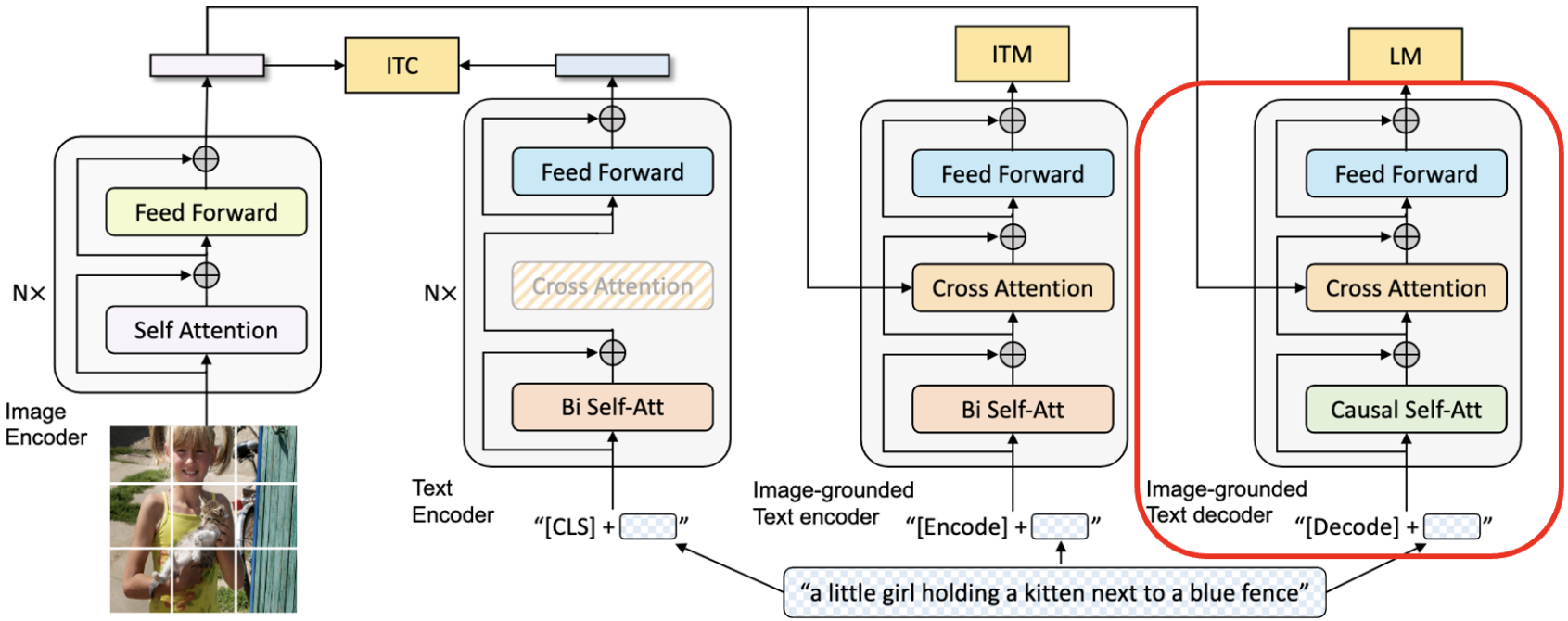
The text encoder is **BERT-base**.

The image-grounded text encoder is modified with **an added cross-attention layer** from the visual encoder between each self-attention and FFN layers.



Approach : decoder

The decoder does cross-attention with the visual encoder + **causal self-attention** (auto-regressive language modeling). Layers are **shared with the text encoder** except for SA.



Approach : recap

This architecture offers quite some flexibility :

- Unimodal encoding : with the image encoder and/or text encoder.
- Image-grounded text encoding.
- Image-grounded text decoding.

Approach : objective

The model is pre-trained with 3 objectives :

- **1 : Image-Text Contrastive loss (ITC)**
- Targets the **unimodal encoders**.
  - **Goal** : to align the features of the vision transformer and the text transformer by encouraging positive image-text pairs to have similar representations in contrast to the negative pairs.
  - **Maths** (from ALBEF presentation) :

$$s(I,T) = g_v(\mathbf{v}_{cls})^T . g'_w(\mathbf{w'}_{cls}),\ s(T,I) = g_w(\mathbf{w}_{cls})^T . g'_v(\mathbf{v'}_{cls})$$

$$p_m^{i2t}(I) = \frac{e^{\frac{s(I,T_m)}{\tau}}}{\Sigma_{m=1}^{M} e^{\frac{s(I,T_k)}{\tau}}},\ p_m^{t2i}(T) = \frac{e^{\frac{s(T,I_m)}{\tau}}}{\Sigma_{k=1}^{M} e^{\frac{s(T,I_k)}{\tau}}}$$

$$\mathcal{L}_{itc} = \frac{1}{2}\mathbb{E}_{(I,T)\sim D}[H(p^{i2t}(I), y^{i2t}(I)) + H(p^{t2i}(T), y^{t2i}(T))]$$

Approach : 1st loss

The model is pre-trained with 3 objectives :

- **2 : Image-Text Matching loss (ITM)**
- Targets the **image-grounded text encoder**.
  - **Goal** : to learn image-text multimodal representation that captures the fine-grained alignment between vision and language.
  - Binary classification task to predict whether an image-text pair is positive (matched) or negative (unmatched) given their multimodal feature.
  - *Following ALBEF* : get negative pairs from batch elements with high contrastive similarity.

Approach : 2nd loss

The model is pre-trained with 3 objectives :

- **3 : Language Modeling loss (LM)**
- Targets the **image-grounded text decoder**.
  - **Goal** : to maximize the likelihood of the text in an autoregressive manner.
  - Label smoothing 0.1 is used.

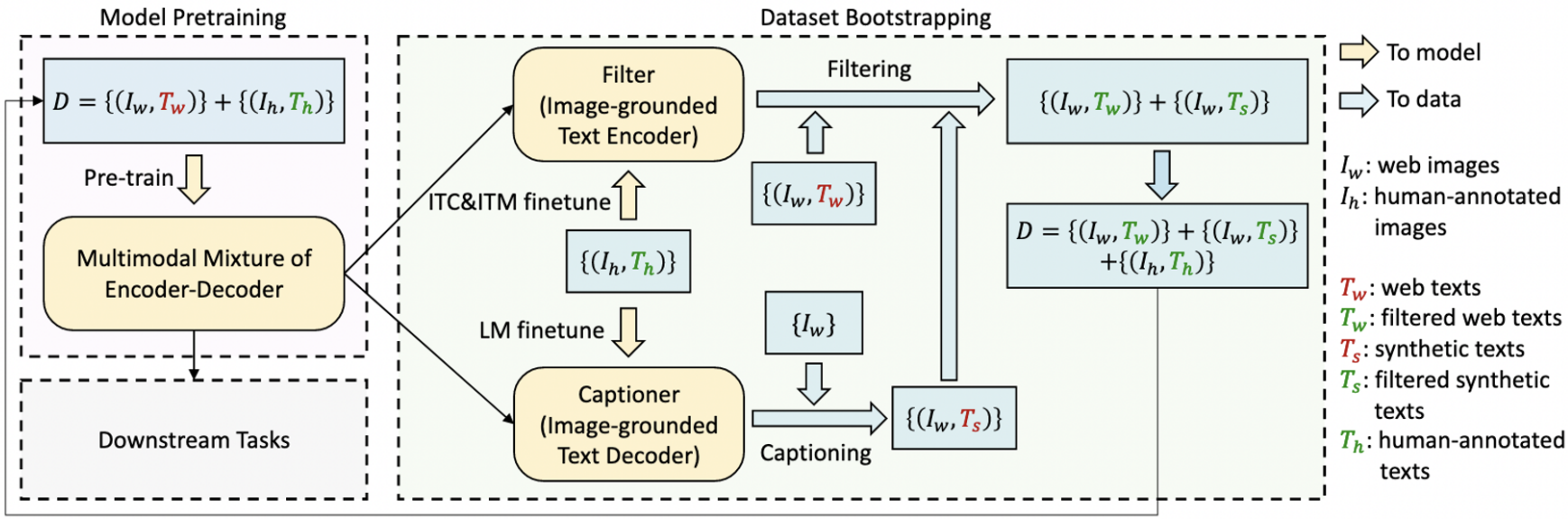
Approach : 3rd loss

The model is pre-trained with 3 objectives :

- **3 : Language Modeling loss (LM)**
- Targets the **image-grounded text decoder**.
  - **Goal** : to maximize the likelihood of the text in an autoregressive manner.
  - Label smoothing 0.1 is used.

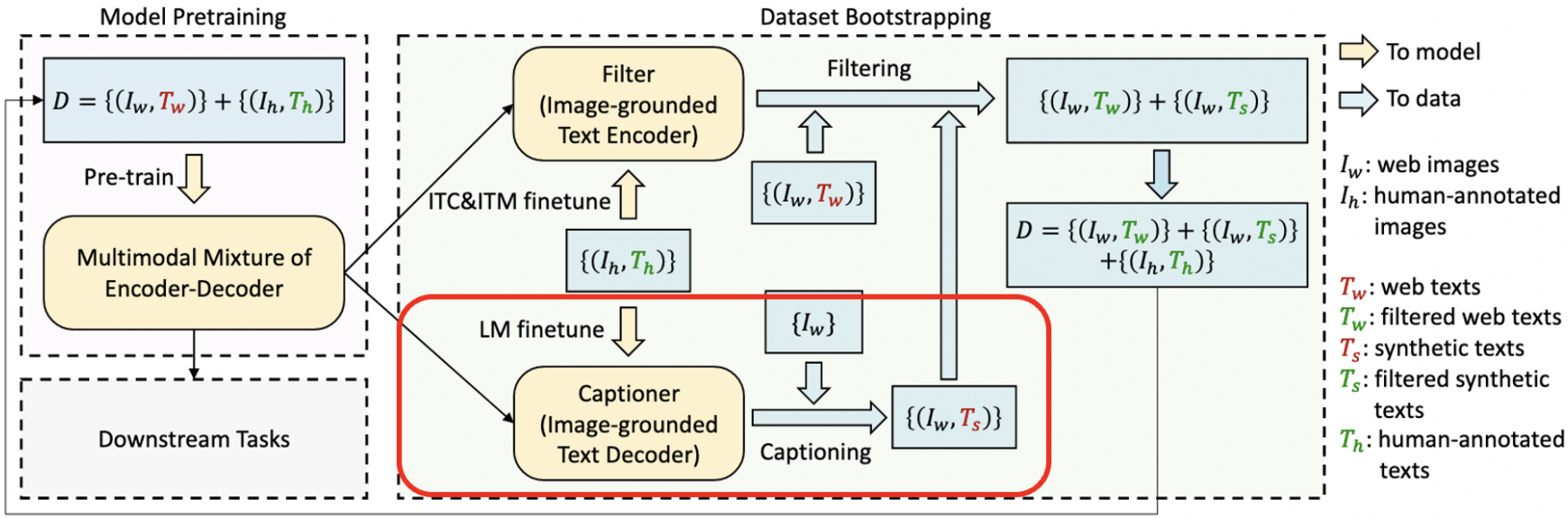
Approach : CapFilt

The other major novelty is **Captioning + Filtering**.
It is initialized with the pre-trained model, and fine-tuned on COCO.
It is used to **create a new dataset for pre-training**.



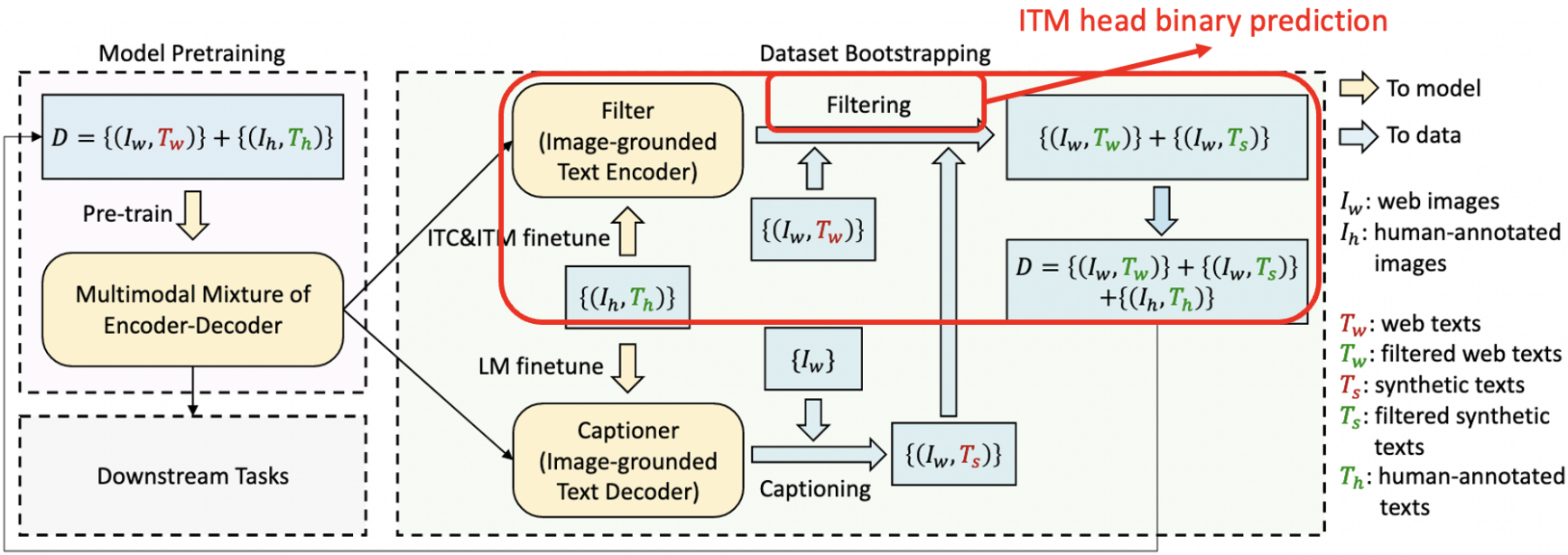
Approach : Captioning

Captioning consists in using the text decoder to **generate a new caption** given a web image.



Approach : Filtering

Filtering consists in using the ITM head to predict **whether to keep image-text pairs** from the (web images, web captions) and (web images, synthetic captions) sets.



Pre-training : Datasets

Same 14M images pre-training dataset as ALBEF.

2 human-annotated datasets :

- COCO : 113k images + 567k texts
- Visual Genome : 100k images + 769k texts

3 web datasets :

- Conceptual Captions : 3M image+text pars
- Conceptual 12M : 10M image+text pairs
- SBU captions : 860k image+text pairs
- (also experimented wih LAION (115M images))

Pre-training : Optimization

- Amazingly pre-training uses only 2 16-GB GPUs.
- Pre-training for 20 epochs.
- AdamW, warming up LR to $3e^{-4}$, decaying linearly at 0.85.

Pre-training : Effect of CapFilt

Each component gives improvement on its own ; and gains are greater when using both.

Pre-train dataset	Bootstrap C	Bootstrap F	Vision backbone	Retrieval-FT (COCO) TR@1	Retrieval-FT (COCO) IR@1	Retrieval-ZS (Flickr) TR@1	Retrieval-ZS (Flickr) IR@1	Caption-FT (COCO) B@4	Caption-FT (COCO) CIDEr	Caption-ZS (NoCaps) CIDEr	Caption-ZS (NoCaps) SPICE
COCO+VG +CC+SBU (14M imgs)	✗	✗	ViT-B/16	78.4	60.7	93.9	82.1	38.0	127.8	102.2	13.9
	✗	✓$_B$		79.1	61.5	94.1	82.8	38.1	128.2	102.7	14.0
	✓$_B$	✗		79.7	62.0	94.4	83.6	38.4	128.9	103.4	14.2
	✓$_B$	✓$_B$		80.6	63.1	94.8	84.9	38.6	129.7	105.1	14.4
COCO+VG +CC+SBU +LAION (129M imgs)	✗	✗	ViT-B/16	79.6	62.0	94.3	83.6	38.8	130.1	105.4	14.2
	✓$_B$	✓$_B$		81.9	64.3	96.0	85.0	39.4	131.4	106.3	14.3
	✓$_L$	✓$_L$		81.2	64.1	96.0	85.5	39.7	133.3	109.6	14.7
	✗	✗	ViT-L/16	80.6	64.1	95.1	85.5	40.3	135.5	112.5	14.7
	✓$_L$	✓$_L$		82.4	65.1	96.7	86.7	40.4	136.7	113.2	14.8

Pre-training : Captioning in CapFilt

Nucleus sampling (aka top-p sampling) is much better than beam search to generate captions.

Generation method	Noise ratio	Retrieval-FT (COCO) TR@1	Retrieval-FT (COCO) IR@1	Retrieval-ZS (Flickr) TR@1	Retrieval-ZS (Flickr) IR@1	Caption-FT (COCO) B@4	Caption-FT (COCO) CIDEr	Caption-ZS (NoCaps) CIDEr	Caption-ZS (NoCaps) SPICE
None	N.A.	78.4	60.7	93.9	82.1	38.0	127.8	102.2	13.9
Beam	19%	79.6	61.9	94.1	83.1	38.4	128.9	103.5	14.2
Nucleus	25%	80.6	63.1	94.8	84.9	38.6	129.7	105.1	14.4

Table 2. Comparison between beam search and nucleus sampling for synthetic caption generation. Models are pre-trained on 14M images.

Nucleus sampling leads to **more diverse** captions, but a **higher noise ratio** too (25% vs 19% for beam search).

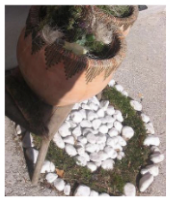
How many captions do they generate per image, and what's the final size of the pre-training dataset after using CapLift ?

Pre-training : Captioning in CapFilt

Examples of captions

T_w: "from bridge near my house"

T_s: "a flock of birds flying over a lake at sunset"



T_w: "in front of a house door in Reichenfels, Austria"

T_s: "a potted plant sitting on top of a pile of rocks"

T_w: "the current castle was built in 1180, replacing a 9th century wooden castle"

T_s: "a large building with a lot of windows on it"

Figure 4. Examples of the web text T_w and the synthetic text T_s. Green texts are accepted by the filter, whereas red texts are rejected.

Fine-tuning : Image-text retrieval

Fine-tune pre-trained model using only **ITC** and **ITM** losses.
First select k candidates based on image-text feature similarity, then re-rank with ITM pair scores.
+2.7% compared to ALBEF (previous SOTA).

Method	Pre-train # Images	COCO (5K test set)						Flickr30K (1K test set)					
		TR			IR			TR			IR		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
UNITER (Chen et al., 2020)	4M	65.7	88.6	93.8	52.9	79.9	88.0	87.3	98.0	99.2	75.6	94.1	96.8
VILLA (Gan et al., 2020)	4M	-	-	-	-	-	-	87.9	97.5	98.8	76.3	94.2	96.8
OSCAR (Li et al., 2020)	4M	70.0	91.1	95.5	54.0	80.8	88.5	-	-	-	-	-	-
UNIMO (Li et al., 2021b)	5.7M	-	-	-	-	-	-	89.4	98.9	99.8	78.0	94.2	97.1
ALIGN (Jia et al., 2021)	1.8B	77.0	93.5	96.9	59.9	83.3	89.8	95.3	99.8	100.0	84.9	97.4	98.6
ALBEF (Li et al., 2021a)	14M	77.6	94.3	97.2	60.7	84.3	90.5	95.9	99.8	100.0	85.6	97.5	98.9
BLIP	14M	80.6	95.2	97.6	63.1	85.3	91.1	96.6	99.8	**100.0**	87.2	97.5	98.8
BLIP	129M	**81.9**	95.4	97.8	**64.3**	85.7	91.5	**97.3**	**99.9**	**100.0**	87.3	97.6	**98.9**
$\text{BLIP}_{\text{CapFilt-L}}$	129M	81.2	**95.7**	**97.9**	64.1	**85.8**	**91.6**	97.2	**99.9**	**100.0**	**87.5**	**97.7**	**98.9**
$\text{BLIP}_{\text{ViT-L}}$	129M	82.4	95.4	97.9	65.1	86.3	91.8	97.4	99.8	99.9	87.6	97.7	99.0

Fine-tuning : Zero-shot image-text retrieval

Even larger improvements in the zero-shot setup.

Method	Pre-train # Images	Flickr30K (1K test set) TR			IR		
		R@1	R@5	R@10	R@1	R@5	R@10
CLIP	400M	88.0	98.7	99.4	68.7	90.6	95.2
ALIGN	1.8B	88.6	98.7	99.7	75.7	93.8	96.8
ALBEF	14M	94.1	99.5	99.7	82.8	96.3	98.1
BLIP	14M	94.8	99.7	**100.0**	84.9	96.7	98.3
BLIP	129M	**96.0**	**99.9**	**100.0**	85.0	**96.8**	98.6
$\text{BLIP}_{\text{CapFilt-L}}$	129M	**96.0**	**99.9**	**100.0**	**85.5**	**96.8**	**98.7**
$\text{BLIP}_{\text{ViT-L}}$	129M	96.7	100.0	100.0	86.7	97.3	98.7

Fine-tuning : Image captioning

Model fine-tuned on COCO with **LM** loss for both datasets.
"A picture of" is used as prompt.

Method	Pre-train #Images	NoCaps validation in-domain C	in-domain S	near-domain C	near-domain S	out-domain C	out-domain S	overall C	overall S	COCO Caption Karpathy test B@4	COCO Caption Karpathy test C
Enc-Dec (Changpinyo et al., 2021)	15M	92.6	12.5	88.3	12.1	94.5	11.9	90.2	12.1	-	110.9
VinVL† (Zhang et al., 2021)	5.7M	103.1	14.2	96.1	13.8	88.3	12.1	95.5	13.5	38.2	129.3
$\text{LEMON}_{\text{base}}$† (Hu et al., 2021)	12M	104.5	14.6	100.7	14.0	96.7	12.4	100.4	13.8	-	-
$\text{LEMON}_{\text{base}}$† (Hu et al., 2021)	200M	107.7	14.7	106.2	14.3	107.9	13.1	106.8	14.1	**40.3**	**133.3**
BLIP	14M	111.3	15.1	104.5	14.4	102.4	13.7	105.1	14.4	38.6	129.7
BLIP	129M	109.1	14.8	105.8	14.4	105.7	13.7	106.3	14.3	39.4	131.4
$\text{BLIP}_{\text{CapFilt-L}}$	129M	**111.8**	**14.9**	**108.6**	**14.8**	**111.5**	**14.2**	**109.6**	**14.7**	39.7	**133.3**
$\text{LEMON}_{\text{large}}$† (Hu et al., 2021)	200M	116.9	15.8	113.3	15.1	111.3	14.0	113.4	15.0	40.6	135.7
$\text{SimVLM}_{\text{huge}}$ (Wang et al., 2021)	1.8B	113.7	-	110.9	-	115.2	-	112.2	-	40.6	143.3
$\text{BLIP}_{\text{ViT-L}}$	129M	114.9	15.2	112.1	14.9	115.3	14.4	113.2	14.8	40.4	136.7

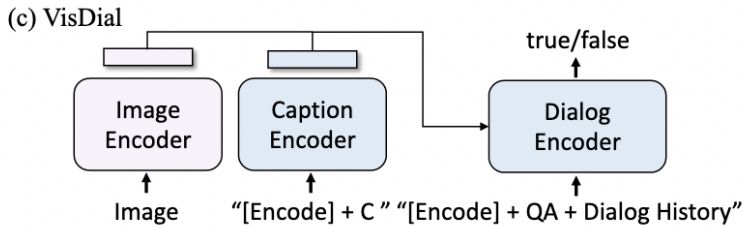
Fine-tuning : Visual question-answering and natural language visual reasoning

Fine-tune with **LM** loss only for VQA.
Add another CA layer to process 2 images for NLVR.

Method	Pre-train #Images	VQA test-dev	VQA test-std	NLVR^2 dev	NLVR^2 test-P
LXMERT	180K	72.42	72.54	74.90	74.50
UNITER	4M	72.70	72.91	77.18	77.85
VL-T5/BART	180K	-	71.3	-	73.6
OSCAR	4M	73.16	73.44	78.07	78.36
SOHO	219K	73.25	73.47	76.37	77.32
VILLA	4M	73.59	73.67	78.39	79.30
UNIMO	5.6M	75.06	75.27	-	-
ALBEF	14M	75.84	76.04	82.55	83.14
SimVLM$_{\text{base}}$†	1.8B	77.87	78.14	81.72	81.77
BLIP	14M	77.54	77.62	**82.67**	82.30
BLIP	129M	78.24	78.17	82.48	**83.08**
BLIP$_{\text{CapFilt-L}}$	129M	**78.25**	**78.32**	82.15	82.24

Fine-tuning : Visual dialog

Adjusting the architecture to encode both the image and the caption.



The dialog encoder is trained to discriminate answers as True/False given the question and the dialog history.

Method	MRR↑	R@1↑	R@5↑	R@10↑	MR↓
VD-BERT	67.44	54.02	83.96	92.33	3.53
VD-ViLBERT†	69.10	55.88	85.50	93.29	3.25
BLIP	**69.41**	**56.44**	**85.90**	**93.30**	**3.20**

Fine-tuning : Zero-shot text-video retrieval

Transfer model fine-tuned on COCO.
Sample n=8 frames uniformly and concatenate their features.

Method	R1↑	R5↑	R10↑	MdR↓
zero-shot				
ActBERT (Zhu & Yang, 2020)	8.6	23.4	33.1	36
SupportSet (Patrick et al., 2021)	8.7	23.0	31.1	31
MIL-NCE (Miech et al., 2020)	9.9	24.0	32.4	29.5
VideoCLIP (Xu et al., 2021)	10.4	22.2	30.0	-
FiT (Bain et al., 2021)	18.7	39.5	51.6	10
BLIP	**43.3**	**65.6**	**74.7**	**2**
finetuning				
ClipBERT (Lei et al., 2021)	22.0	46.8	59.9	6
VideoCLIP (Xu et al., 2021)	30.9	55.4	66.8	-

No temporal information, but it still reaches zero-shot SOTA !

Fine-tuning : Zero-shot video question-answering

Transfer model fine-tuned on VQA.
Sample n=16 frames uniformly and concatenate their features.

Method	MSRVTT-QA	MSVD-QA
zero-shot		
VQA-T (Yang et al., 2021)	2.9	7.5
BLIP	19.2	35.2
finetuning		
HME (Fan et al., 2019)	33.0	33.7
HCRN (Le et al., 2020)	35.6	36.1
VQA-T (Yang et al., 2021)	41.5	46.3

No temporal information, but it still reaches zero-shot SOTA !

Conclusion

- Elegant follow-up work to ALBEF
- Impressive results !
- CapLift is an elegant way to leverage the pre-trained and partially fine-tuned model to clean web data.
  - Can we use a similar approach in text (only) pre-training objectives ?
  - Some more info on how best to use CapLift : how many captions per image, probability threshold for ITM prediction, etc would be appreciated.