Vision-Language Pre-training: Part I

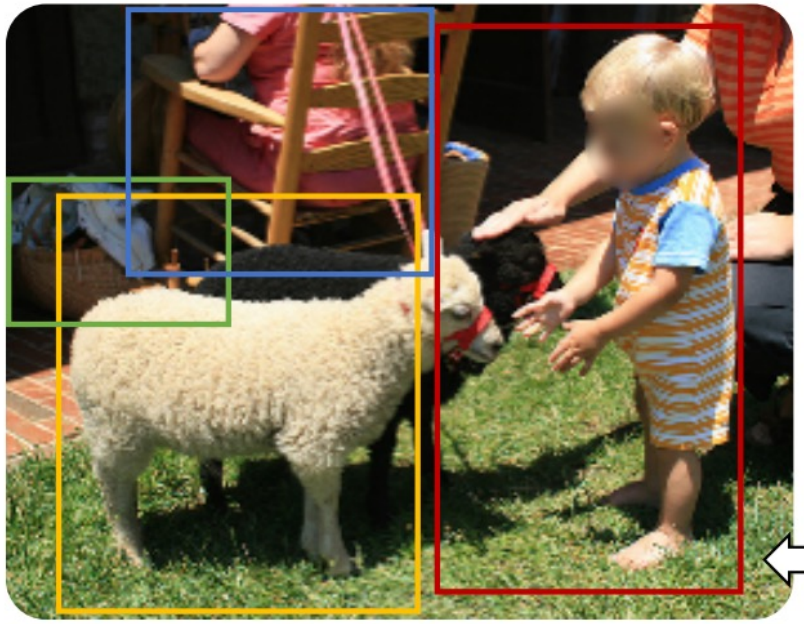
Zhe Gan

Senior Researcher

Microsoft

What is Vision-Language Research?

- How to train a smart AI system that has joint understanding of both image/video and text modalities
- *Popular tasks:*



Visual Question Answering

What color is the child's outfit? Orange

Referring Expressions

child sheep basket people sitting on chair

Multi-modal Verification

The child is petting a dog. false

Caption-based Image Retrieval and *image captioning*

A child in orange clothes plays with sheep.

Image from: https://arxiv.org/pdf/1912.02315.pdf

Why Vision and Language?

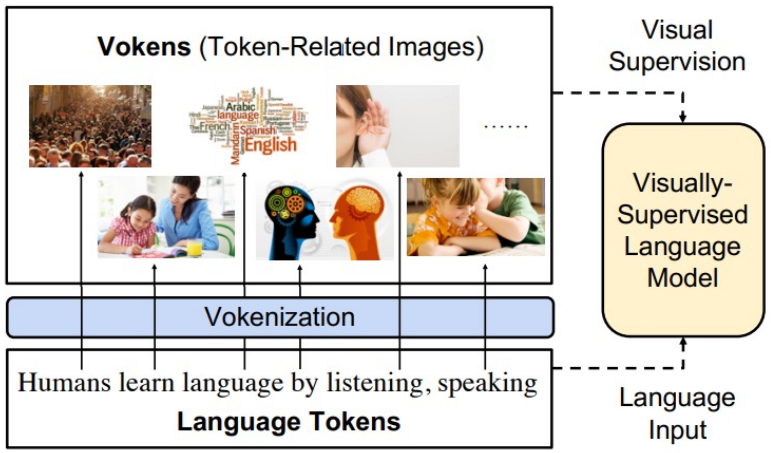
- This is how we interact with and learn from the world
  - *Vision* is a large portion of how humans perceive
  - *Language* is a large portion of how humans communicate
  - A smart AI system should be able to perform well on both
  - Great potential for visually impaired people

- *VL for V*: A scalable way to learn visual representations
  - SOTA computer vision models rely on carefully annotated labels/bounding boxes for supervised learning
  - Self-supervised learning is scalable, but supervision signal can be weak
  - Image-text pairs widely exist on the web
    - OpenAI CLIP[1] /Google ALIGN[2] uses 400M/1.8B pairs for model training

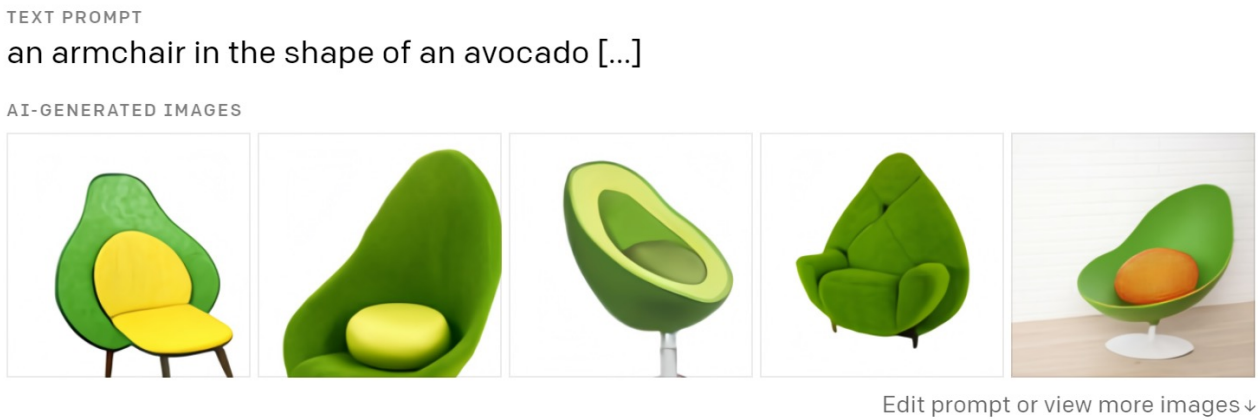
[1] Learning Transferable Visual Models From Natural Language Supervision, 2021
[2] Scaling Up Visual and Vision-Language Representation Learning With Noisy Text Supervision, ICML 2021

Why Vision and Language?

- *VL for L*: Great potential to enhance language representations
  - A picture is worth a thousand words
  - Vokenization, EMNLP 2020
  - Video-aided Unsupervised Grammar Induction (NAACL 2021 Best Long Paper)



- *It is just fun!*
  - OpenAI DALL-E[1]: Creating images from text



[1] https://openai.com/blog/dall-e/

Vision and Language in CVPR 2020

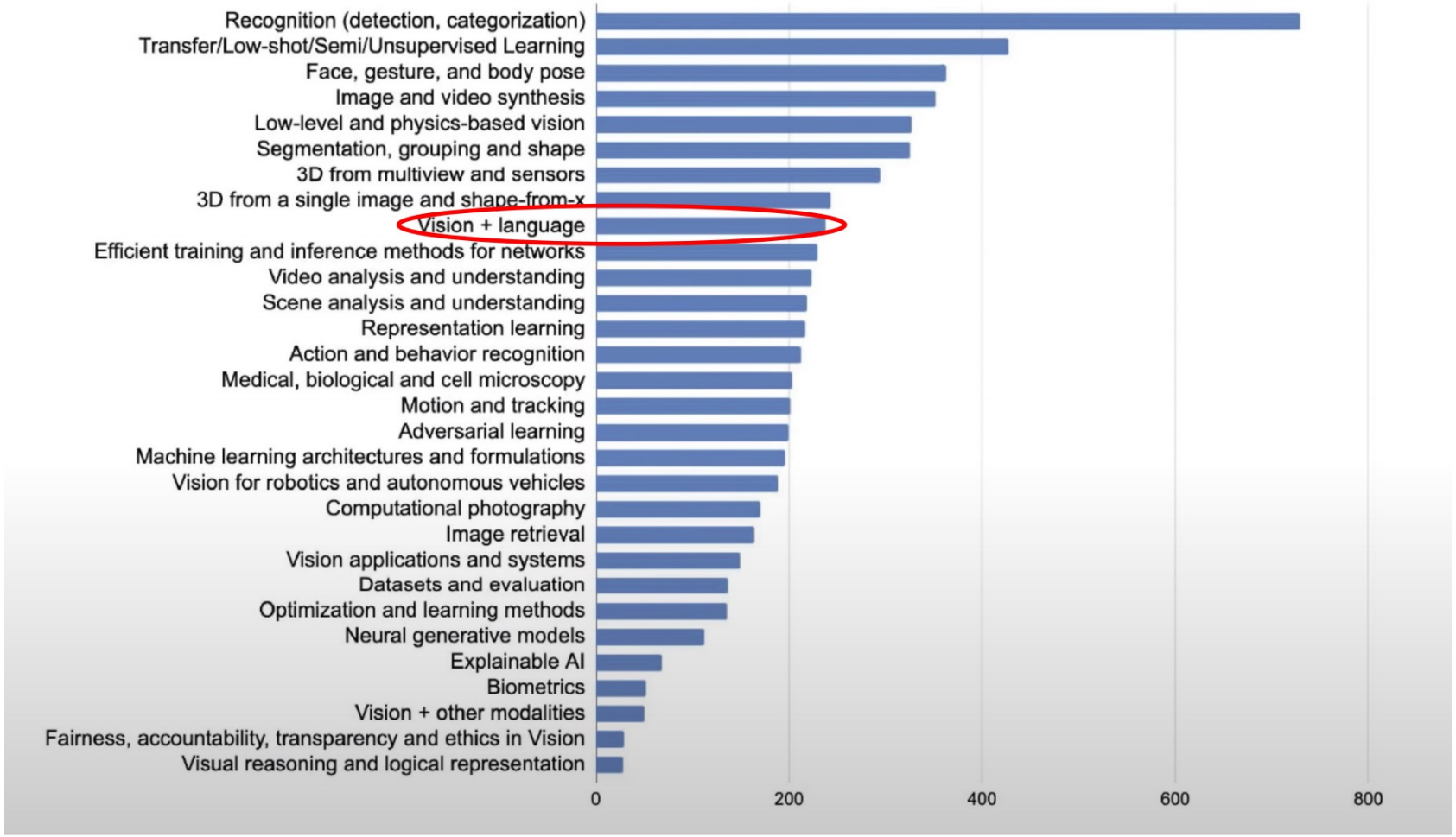


Image from: https://alvr-workshop.github.io/2020/talks/ALVR_opening.pdf

Vision and Language in ACL 2020

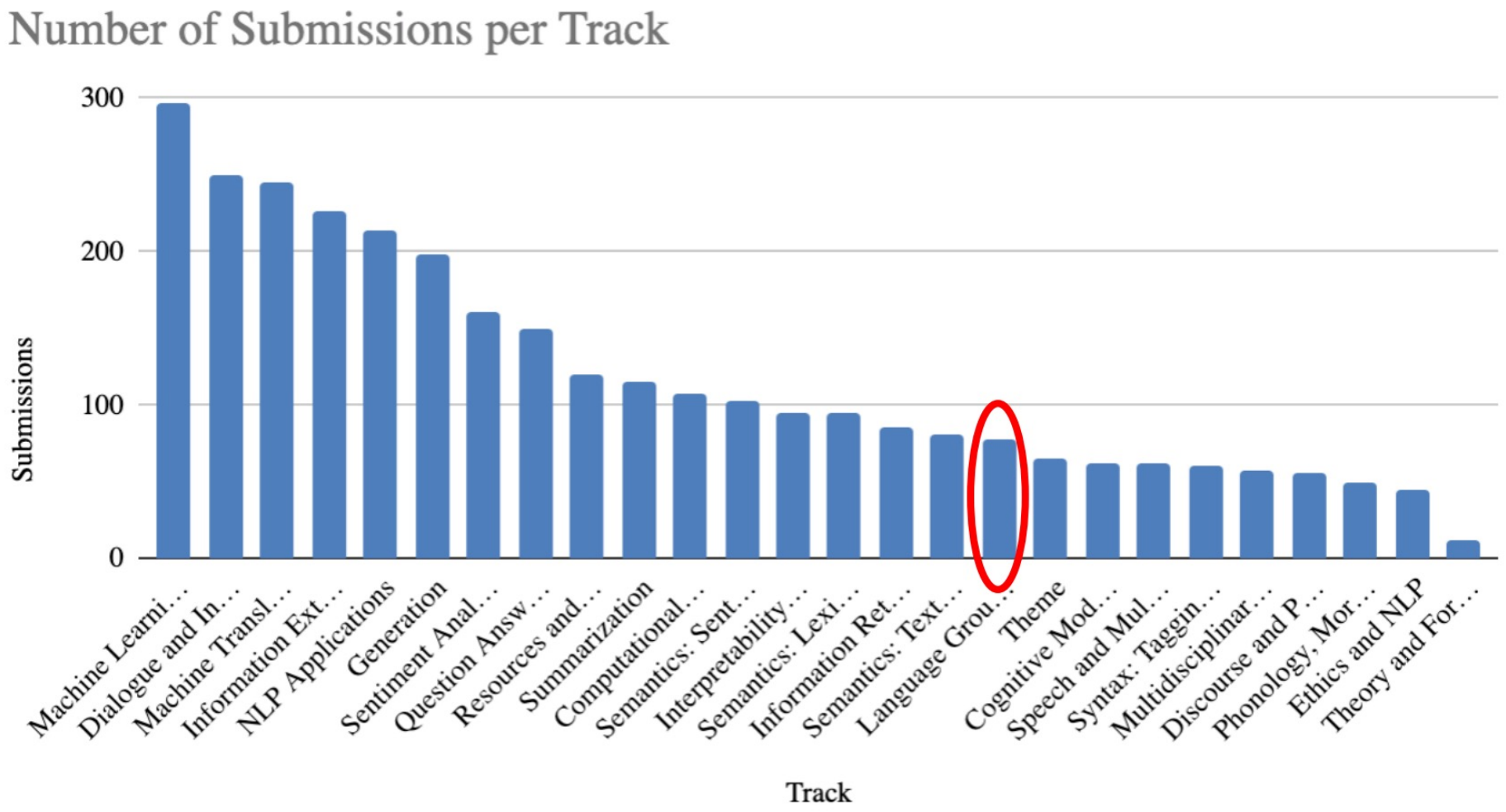


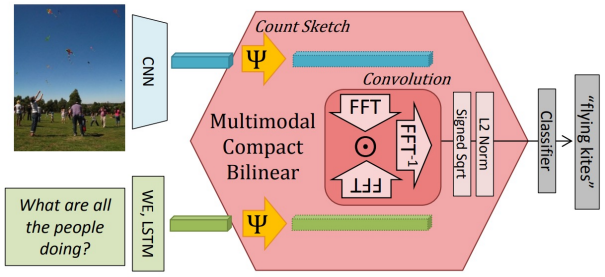
Image from: https://alvr-workshop.github.io/2020/talks/ALVR_opening.pdf

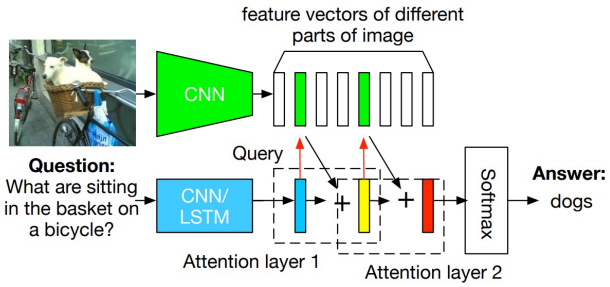
How to Perform VL Research?

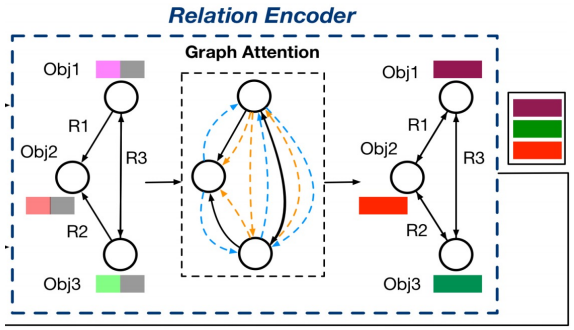
- Before 2019:
  - Bilinear pooling
  - All kinds of attention
  - Incorporation of object relations
  - Multi-step reasoning
  - Neural modules

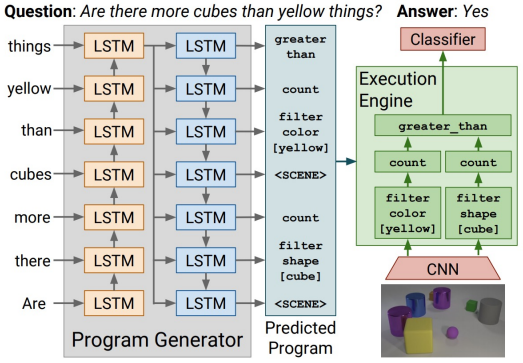
Tutorial at CVPR 2020

1:15 - 1:25	**Opening Remarks** presented by JJ Liu and Xiaodong He (Slides , YouTube , Bilibili)
1:25 - 2:15	**Visual QA and Reasoning** presented by Zhe Gan (Slides , YouTube , Bilibili)
2:15 - 2:30	**Coffee Break**
2:30 - 3:10	**Visual Captioning** presented by Luowei Zhou (Slides , YouTube , Bilibili)
3:10 - 3:40	**Text-to-image Synthesis** presented by Yu Cheng (Slides , YouTube , Bilibili)
3:40 - 4:00	**Coffee Break**
4:00 - 5:00	**Self-supervised Learning** presented by Licheng Yu , Linjie Li and Yen-Chun Chen (Slides



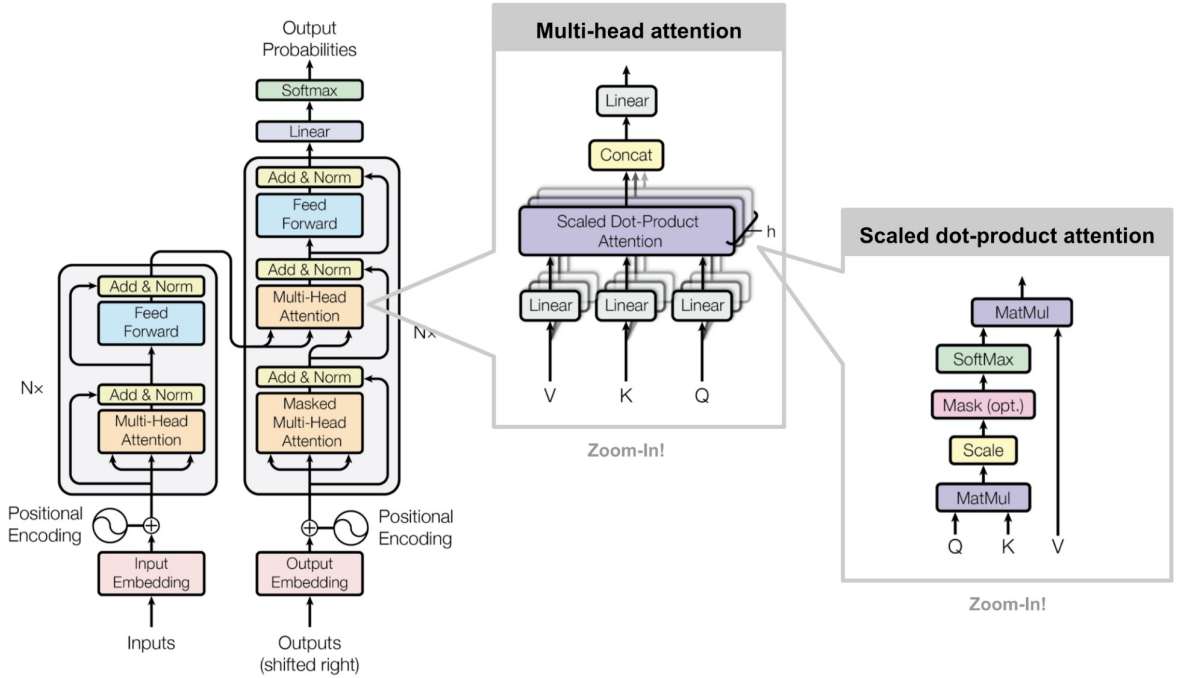






How to Perform VL Research?

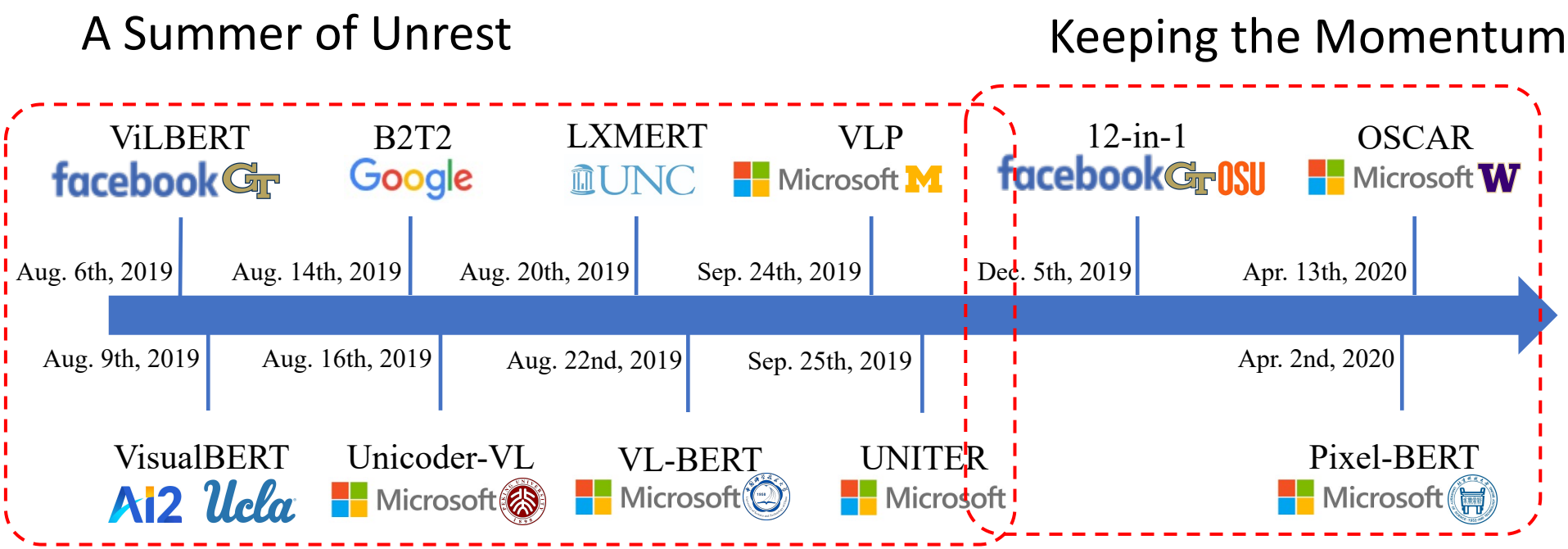
- Before 2019:
  - Bilinear pooling
  - All kinds of attention
  - Incorporation of object relations
  - Multi-step reasoning
  - Neural modules



- After 2019:
  - Large-scale *transformer*-based *self-supervised* pre-training

- *Transformer:* first proposed for NLP, popularized by BERT and GPT-2/3, extended to image generation, vision-language pre-training, and now image classification

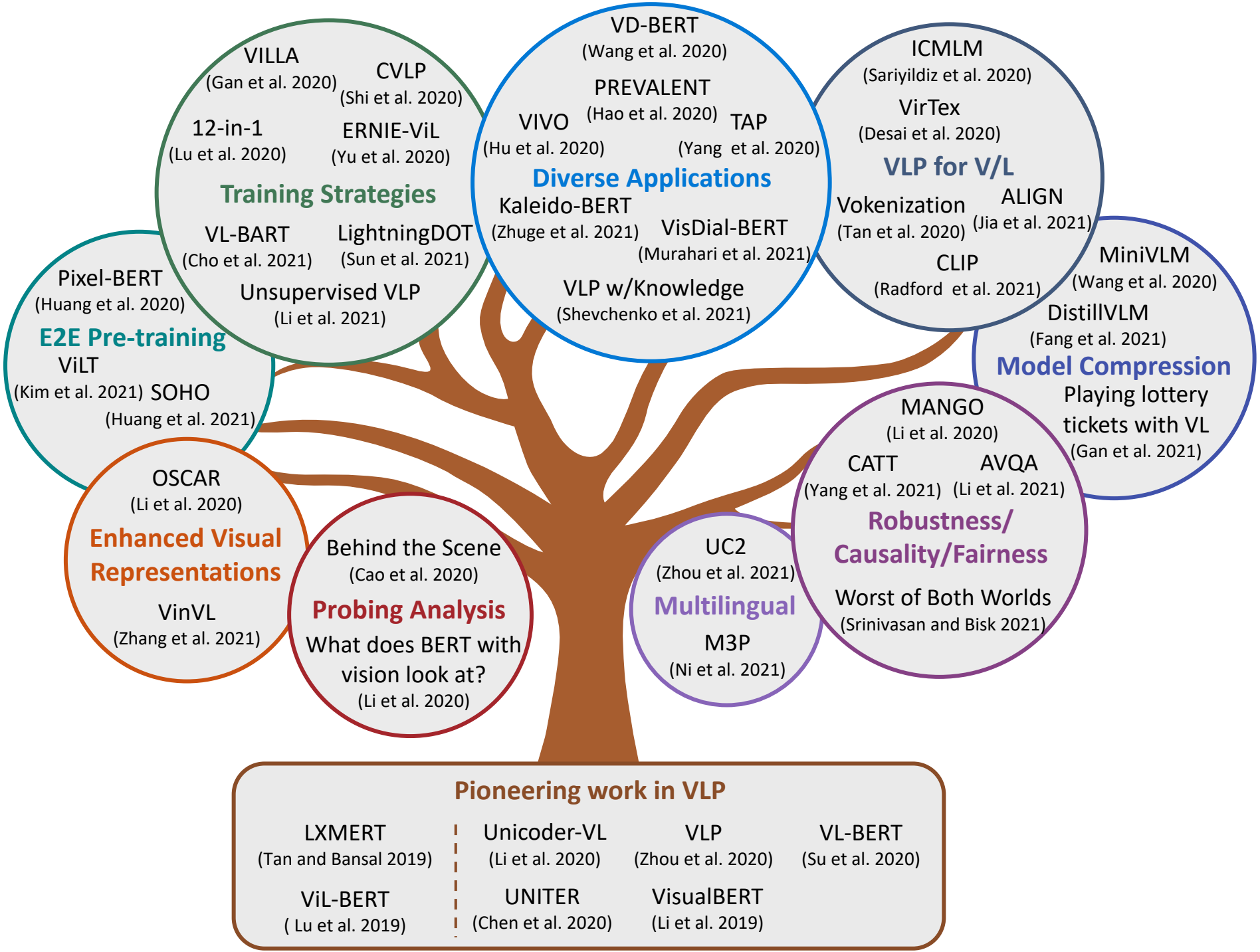
Great success of VLP models

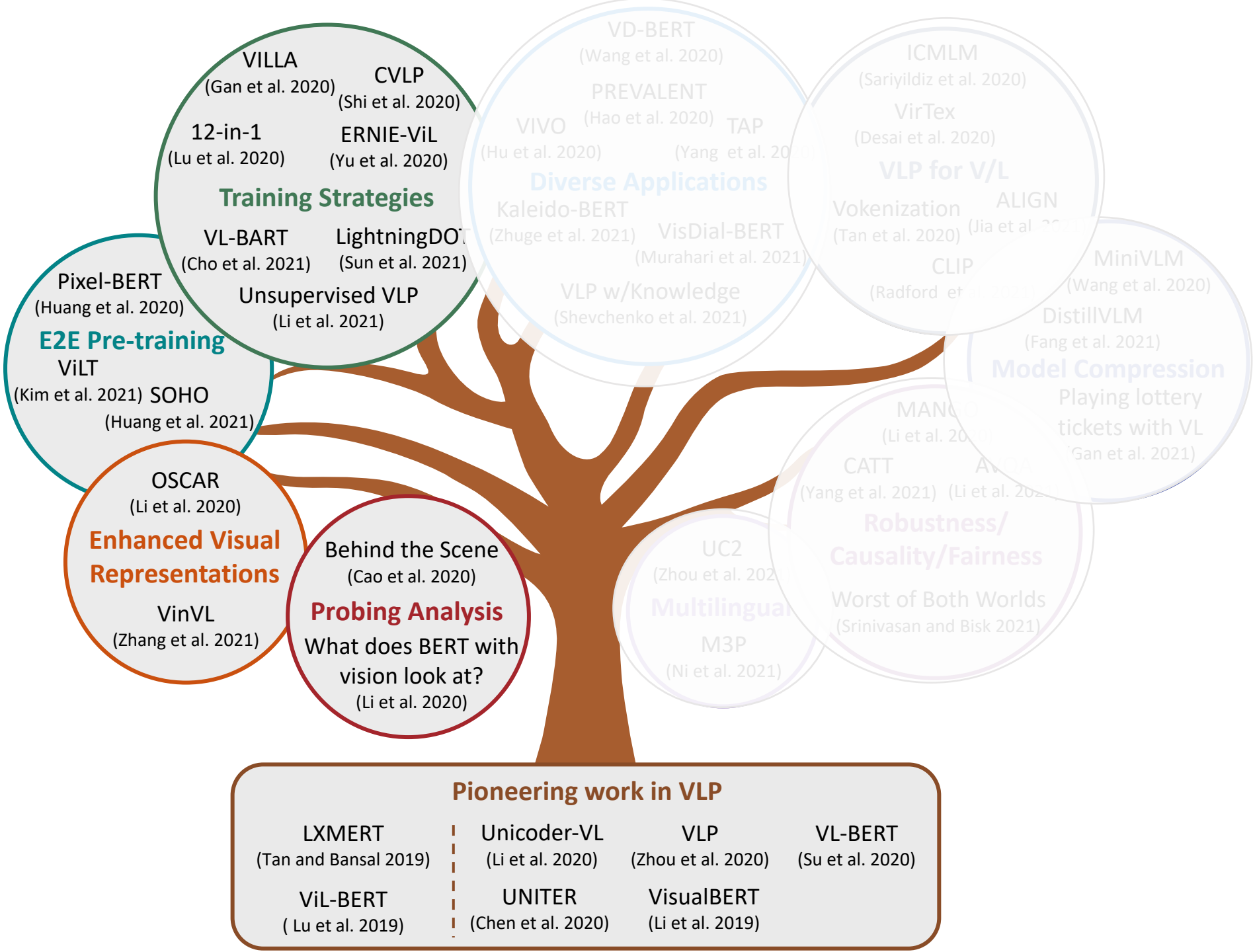


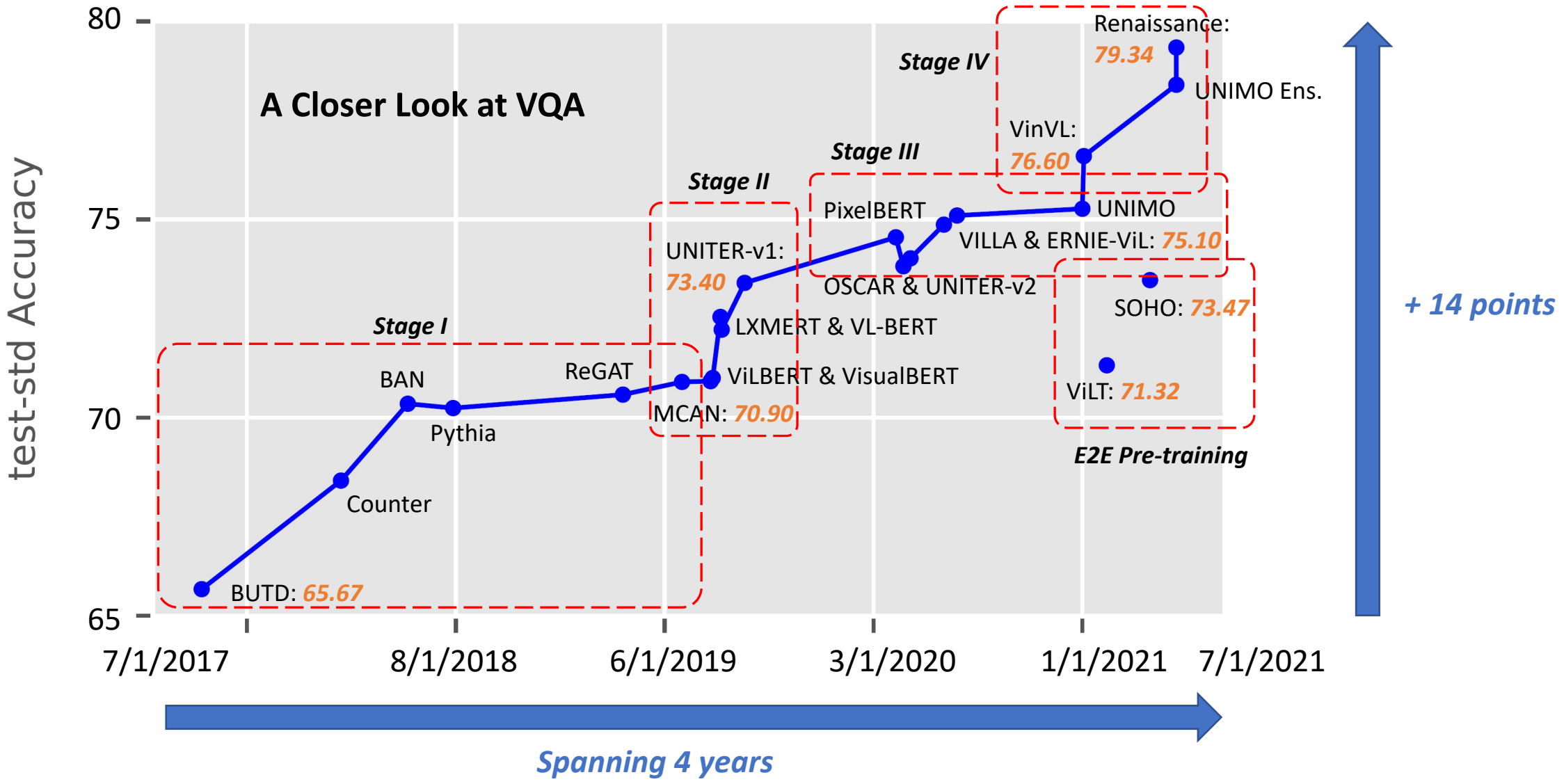
Downstream Tasks

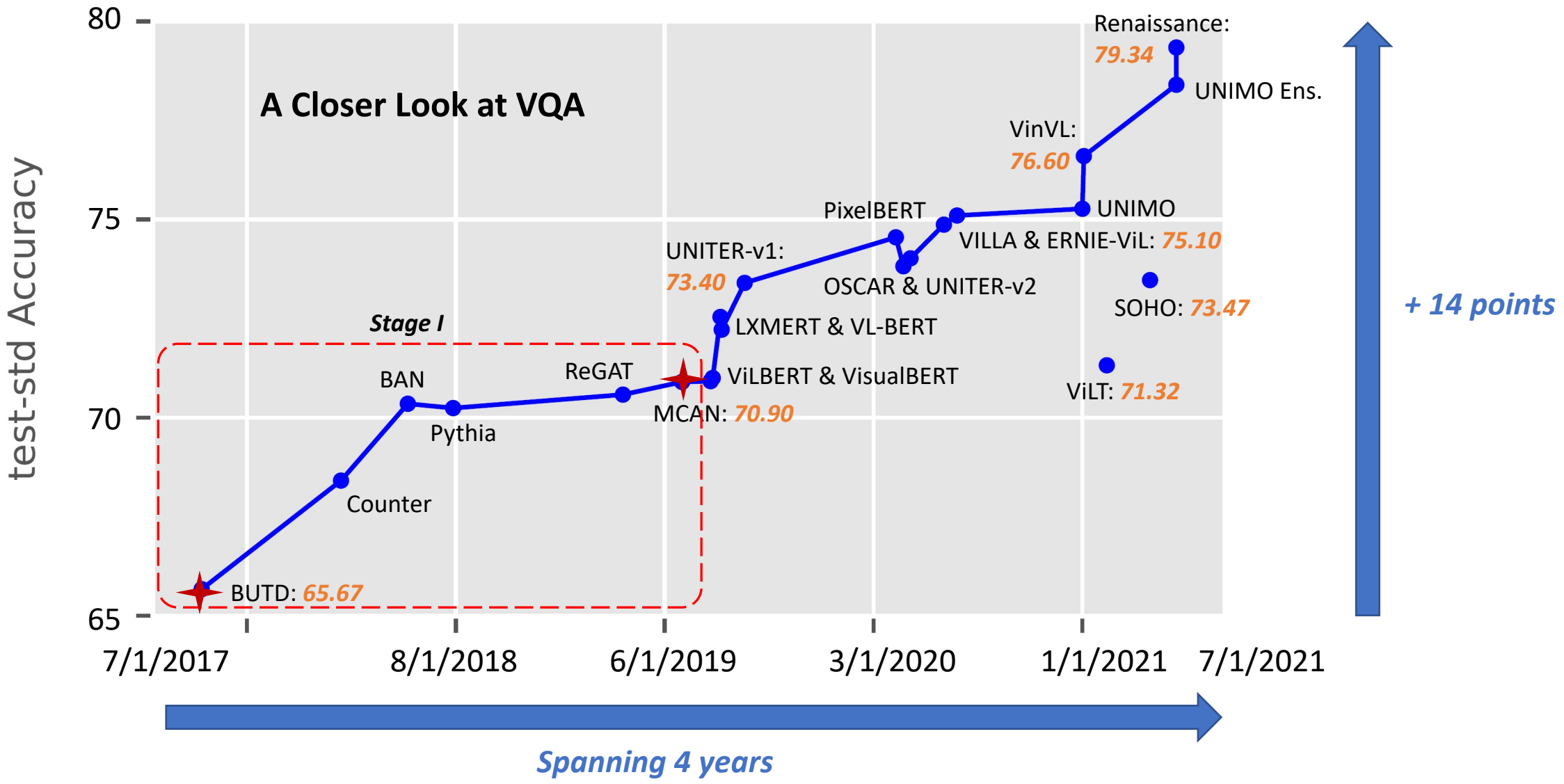
- VQA
- VCR
- NLVR2
- Visual Entailment
- Referring Expressions
- Image-Text Retrieval
- Image Captioning

Many more models have been proposed since then


VILLA
(Gan et al. 2020)
CVLP
(Shi et al. 2020)
12-in-1
(Lu et al. 2020)
ERNIE-ViL
(Yu et al. 2020)
Training Strategies
VL-BART
(Cho et al. 2021)
LightningDOT
(Sun et al. 2021)
Unsupervised VLP
(Li et al. 2021)
VD-BERT
(Wang et al. 2020)
PREVALENT
(Hao et al. 2020)
VIVO
(Hu et al. 2020)
TAP
(Yang et al. 2020)
Diverse Applications
Kaleido-BERT
(Zhuge et al. 2021)
VisDial-BERT
(Murahari et al. 2021)
VLP w/Knowledge
(Shevchenko et al. 2021)
ICMLM
(Sariyildiz et al. 2020)
VirTex
(Desai et al. 2020)
VLP for V/L
Vokenization
(Tan et al. 2020)
ALIGN
(Jia et al. 2021)
CLIP
(Radford et al. 2021)
Pixel-BERT
(Huang et al. 2020)
E2E Pre-training
ViLT
(Kim et al. 2021)
SOHO
(Huang et al. 2021)
MiniVLM
(Wang et al. 2020)
DistillVLM
(Fang et al. 2021)
Model Compression
Playing lottery tickets with VL
(Gan et al. 2021)
OSCAR
(Li et al. 2020)
Enhanced Visual Representations
VinVL
(Zhang et al. 2021)
Behind the Scene
(Cao et al. 2020)
Probing Analysis
What does BERT with vision look at?
(Li et al. 2020)
UC2
(Zhou et al. 2021)
Multilingual
M3P
(Ni et al. 2021)
MANGO
(Li et al. 2020)
CATT
(Yang et al. 2021)
AVQA
(Li et al. 2021)
Robustness/ Causality/Fairness
Worst of Both Worlds
(Srinivasan and Bisk 2021)
Pioneering work in VLP
LXMERT
(Tan and Bansal 2019)
Unicoder-VL
(Li et al. 2020)
VLP
(Zhou et al. 2020)
VL-BERT
(Su et al. 2020)
ViL-BERT
(Lu et al. 2019)
UNITER
(Chen et al. 2020)
VisualBERT
(Li et al. 2019)


VILLA
(Gan et al. 2020)
CVLP
(Shi et al. 2020)
12-in-1
(Lu et al. 2020)
ERNIE-ViL
(Yu et al. 2020)
Training Strategies
VL-BART
(Cho et al. 2021)
LightningDOT
(Sun et al. 2021)
Unsupervised VLP
(Li et al. 2021)
Pixel-BERT
(Huang et al. 2020)
E2E Pre-training
ViLT
(Kim et al. 2021)
SOHO
(Huang et al. 2021)
OSCAR
(Li et al. 2020)
Enhanced Visual Representations
VinVL
(Zhang et al. 2021)
Behind the Scene
(Cao et al. 2020)
Probing Analysis
What does BERT with vision look at?
(Li et al. 2020)
VD-BERT
(Wang et al. 2020)
PREVALENT
(Hao et al. 2020)
VIVO
(Hu et al. 2020)
TAP
Diverse Applications
Kaleido-BERT
(Zhuge et al. 2021)
VisDial-BERT
(Murahari et al. 2021)
VLP w/Knowledge
(Shevchenko et al. 2021)
ICMLM
(Sariyildiz et al. 2020)
VirTex
(Desai et al. 2020)
VLP for V/L
Vokenization
(Tan et al. 2020)
ALIGN
CLIP
MiniVLM
(Wang et al. 2020)
DistillVLM
(Fang et al. 2021)
Model Compression
Playing lottery tickets with VL
Gan et al. 2021)
CATT
(Yang et al. 2021)
Robustness/ Causality/Fairness
Worst of Both Worlds
(Srinivasan and Bisk 2021)
UC2
Multilingual
M3P
(Ni et al. 2021)
Pioneering work in VLP
LXMERT
(Tan and Bansal 2019)
Unicoder-VL
(Li et al. 2020)
VLP
(Zhou et al. 2020)
VL-BERT
(Su et al. 2020)
ViL-BERT
(Lu et al. 2019)
UNITER
(Chen et al. 2020)
VisualBERT
(Li et al. 2019)


A Closer Look at VQA
test-std Accuracy
80
75
70
65
7/1/2017
8/1/2018
6/1/2019
3/1/2020
1/1/2021
7/1/2021
Stage I
Stage II
Stage III
Stage IV
E2E Pre-training
BUTD: 65.67
Counter
BAN
Pythia
ReGAT
MCAN: 70.90
UNITER-v1: 73.40
ViLBERT & VisualBERT
LXMERT & VL-BERT
PixelBERT
OSCAR & UNITER-v2
VILLA & ERNIE-ViL: 75.10
UNIMO
VinVL: 76.60
ViLT: 71.32
SOHO: 73.47
Renaissance: 79.34
UNIMO Ens.
+ 14 points
Spanning 4 years


A Closer Look at VQA
test-std Accuracy
80
75
70
65
7/1/2017
8/1/2018
6/1/2019
3/1/2020
1/1/2021
7/1/2021
Stage I
BUTD: 65.67
Counter
BAN
Pythia
ReGAT
MCAN: 70.90
ViLBERT & VisualBERT
LXMERT & VL-BERT
UNITER-v1: 73.40
OSCAR & UNITER-v2
PixelBERT
VILLA & ERNIE-ViL: 75.10
UNIMO
VinVL: 76.60
ViLT: 71.32
SOHO: 73.47
Renaissance: 79.34
UNIMO Ens.
+ 14 points
Spanning 4 years

How a typical VQA system works

Bottom Up and Top Down Attention

- 2017 VQA Challenge Winner
  - The use of object-centric image representations

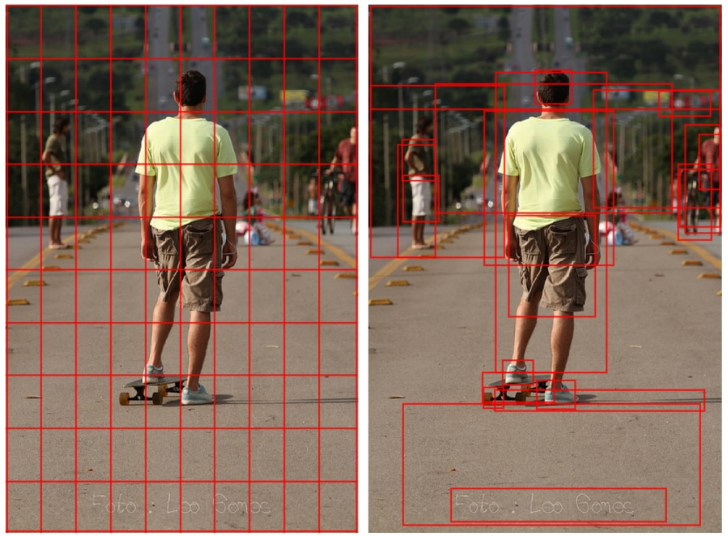
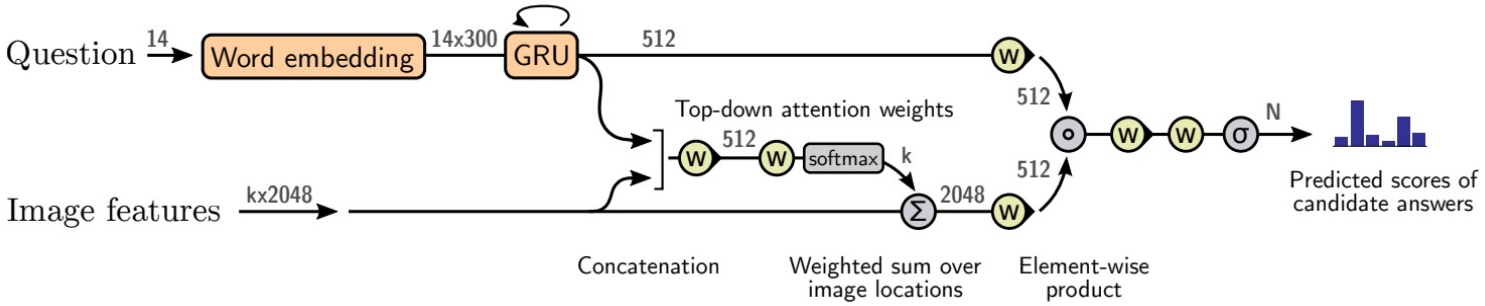


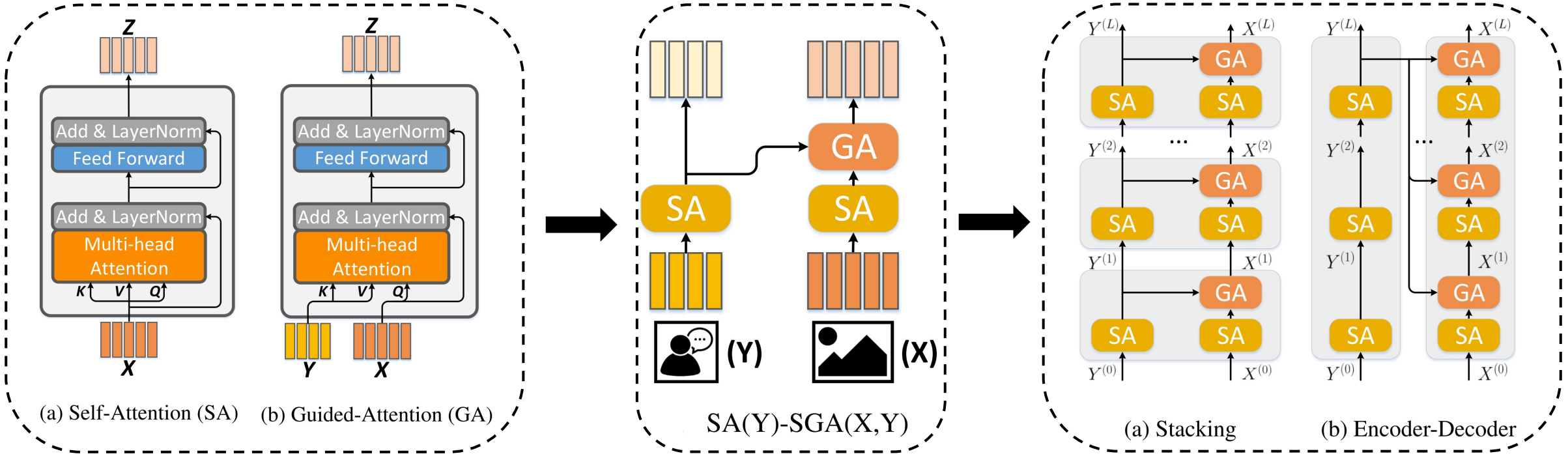
Figure 1. Typically, attention models operate on CNN features corresponding to a uniform grid of equally-sized image regions (left). Our approach enables attention to be calculated at the level of objects and other salient image regions (right).



[1] Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering, CVPR 2018

MCAN: Deep Modular Co-Attention Network

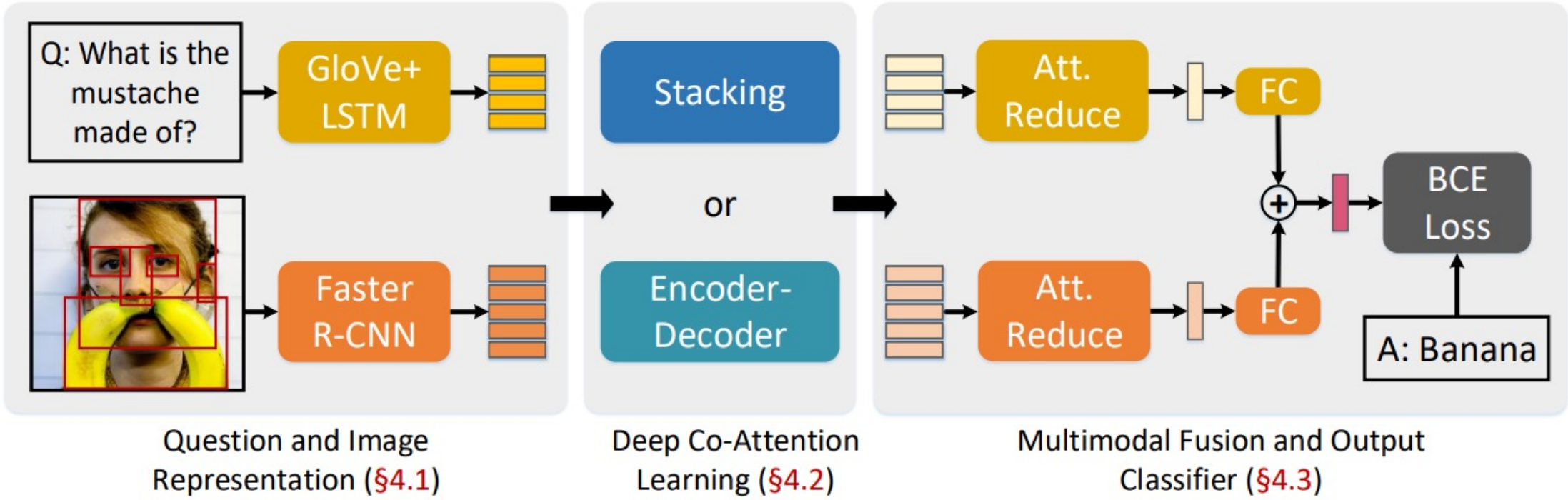
- Winning entry to VQA Challenge 2019, already close to *vision-language pre-training* models



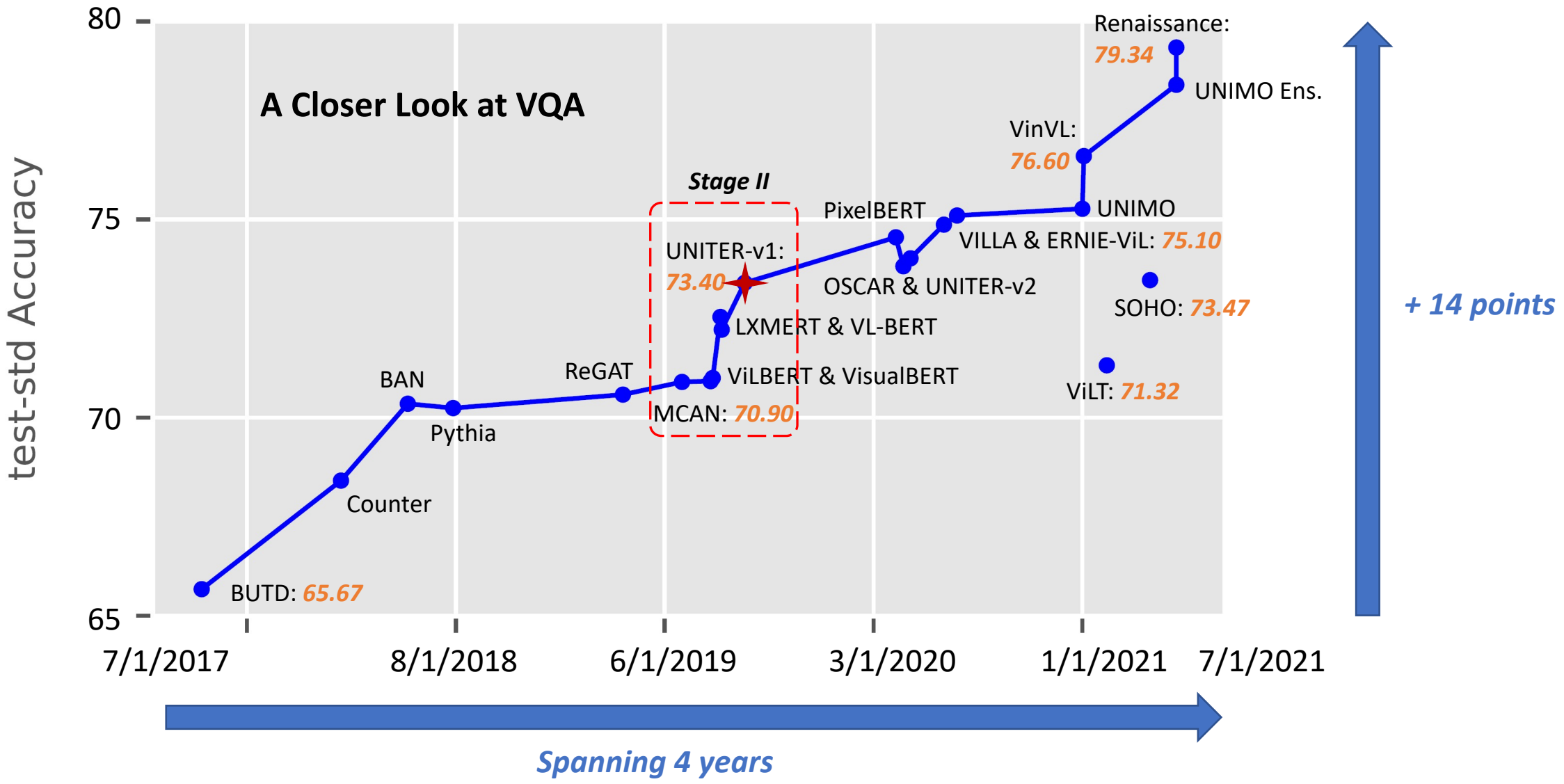
[1] Deep Modular Co-Attention Networks for Visual Question Answering, CVPR 2019
[2] Dynamic Fusion with Intra- and Inter- Modality Attention Flow for Visual Question Answering, CVPR 2019

MCAN: Deep Modular Co-Attention Network

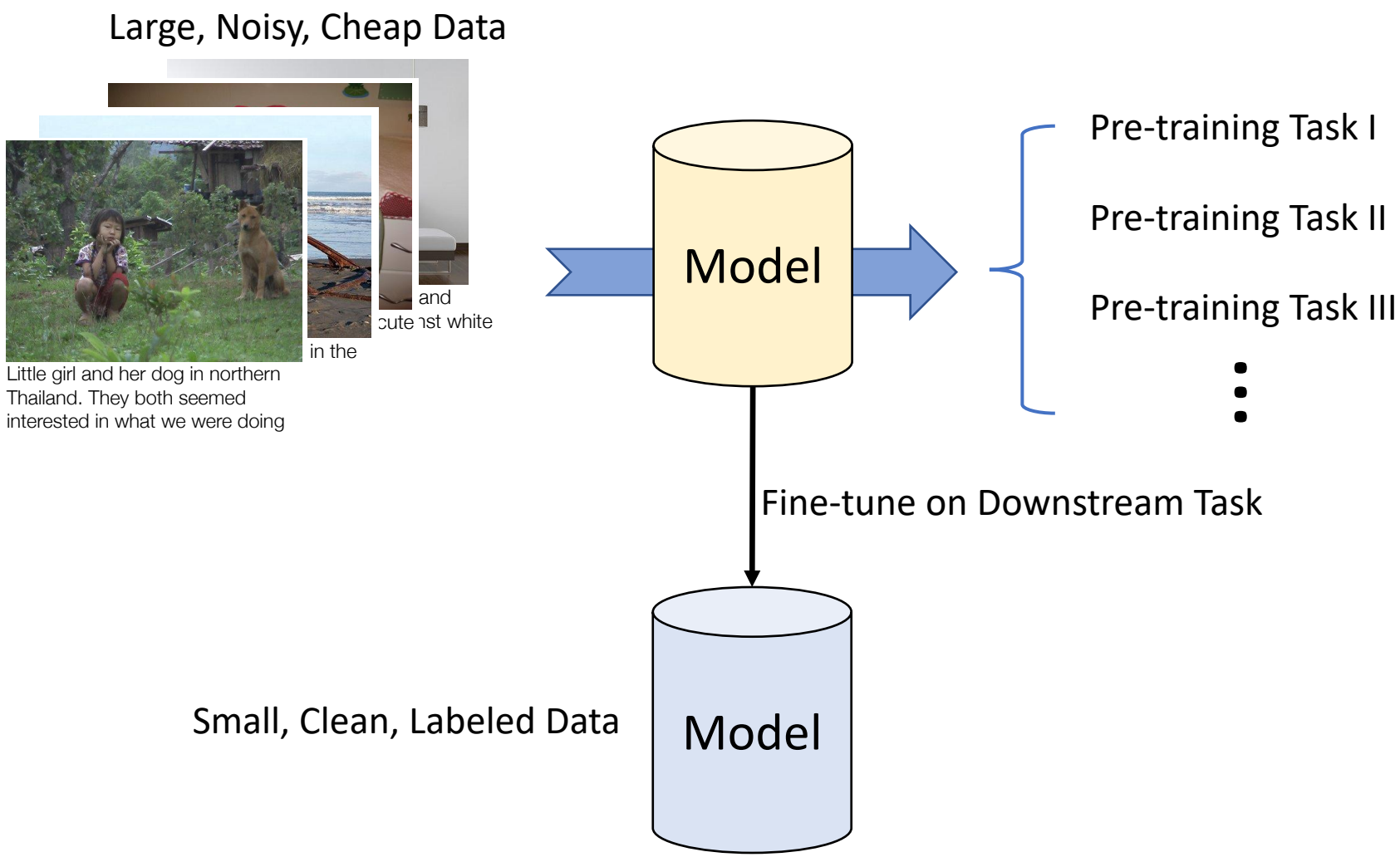
- Winning entry to VQA Challenge 2019, already close to *vision-language pre-training* models



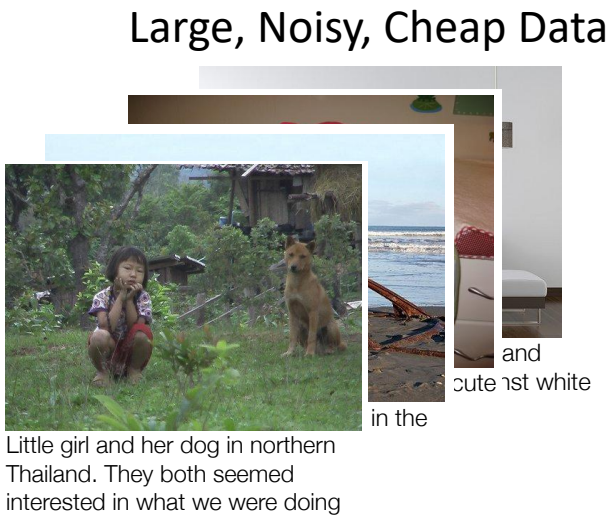
[1] Deep Modular Co-Attention Networks for Visual Question Answering, CVPR 2019
[2] Dynamic Fusion with Intra- and Inter- Modality Attention Flow for Visual Question Answering, CVPR 2019


A Closer Look at VQA
test-std Accuracy
80
75
70
65
7/1/2017
8/1/2018
6/1/2019
3/1/2020
1/1/2021
7/1/2021
BUTD: 65.67
Counter
BAN
Pythia
ReGAT
MCAN: 70.90
Stage II
UNITER-v1: 73.40
ViLBERT & VisualBERT
LXMERT & VL-BERT
OSCAR & UNITER-v2
PixelBERT
VILLA & ERNIE-ViL: 75.10
UNIMO
VinVL: 76.60
ViLT: 71.32
SOHO: 73.47
Renaissance: 79.34
UNIMO Ens.
+ 14 points
Spanning 4 years

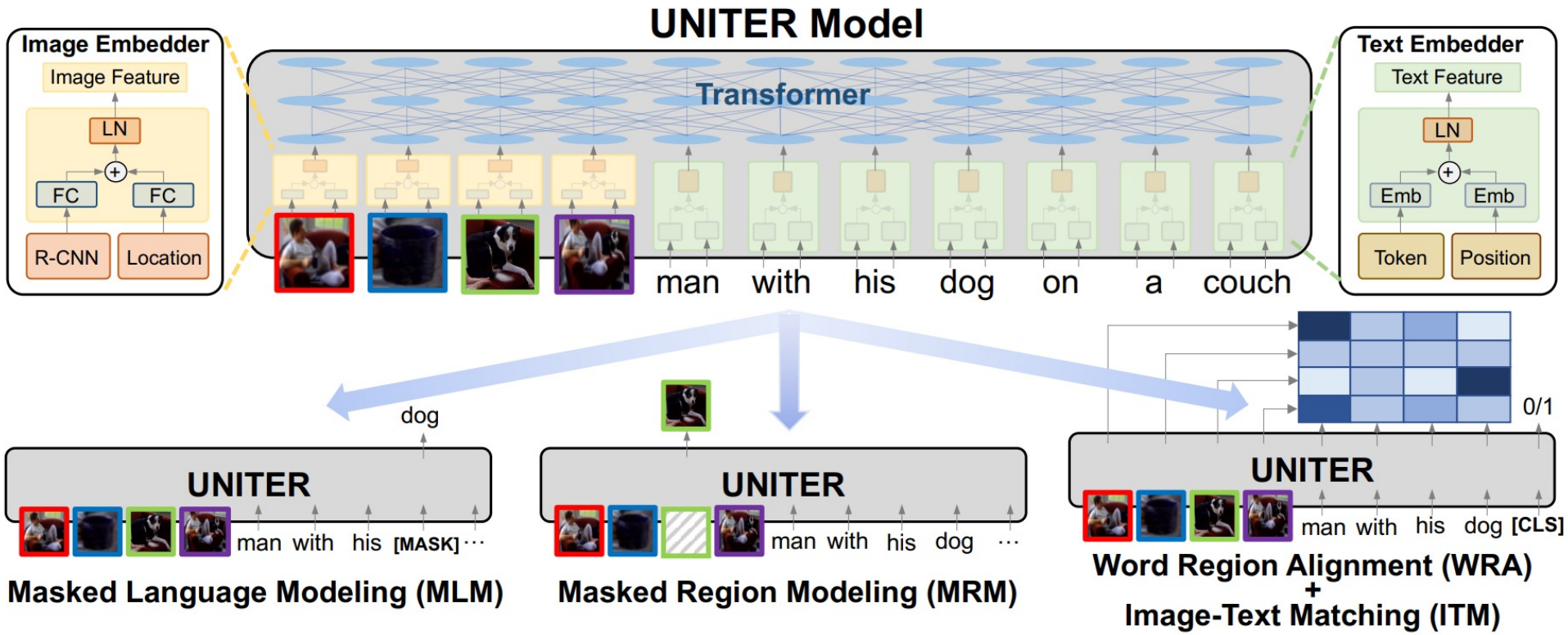
Two-Stage Training Pipeline



Generalization



UNITER Framework



[1] UNITER: UNiversal Image-TExt Representation Learning, ECCV 2020

Common Pre-training Data for Vision + Language

	In-domain		Out-of-domain	
Split	COCO Captions	VG Dense Captions	Conceptual Captions	SBU Captions
train	533K (106K)	5.06M (101K)	3.0M (3.0M)	990K (990K)
val	25K (5K)	106K (2.1K)	14K (14K)	10K (10K)

Conceptual Caption

Alt-text: A Pakistani worker helps to clear the debris from the Taj Mahal Hotel November 7, 2005 in Balakot, Pakistan.

Conceptual Captions: a worker helps to clear the debris.

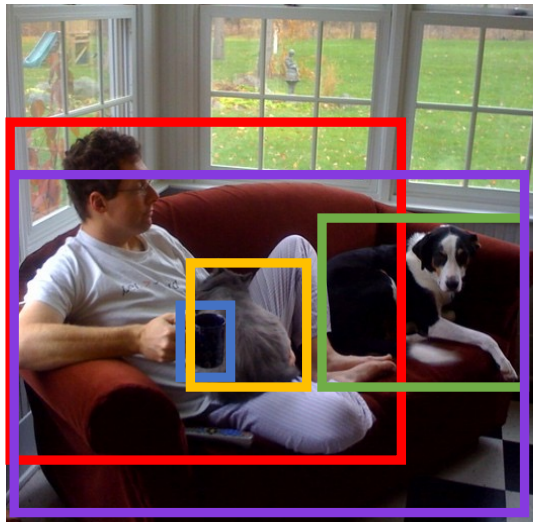
SBU Caption

Little girl and her dog in northern Thailand. They both seemed interested in what we were doing

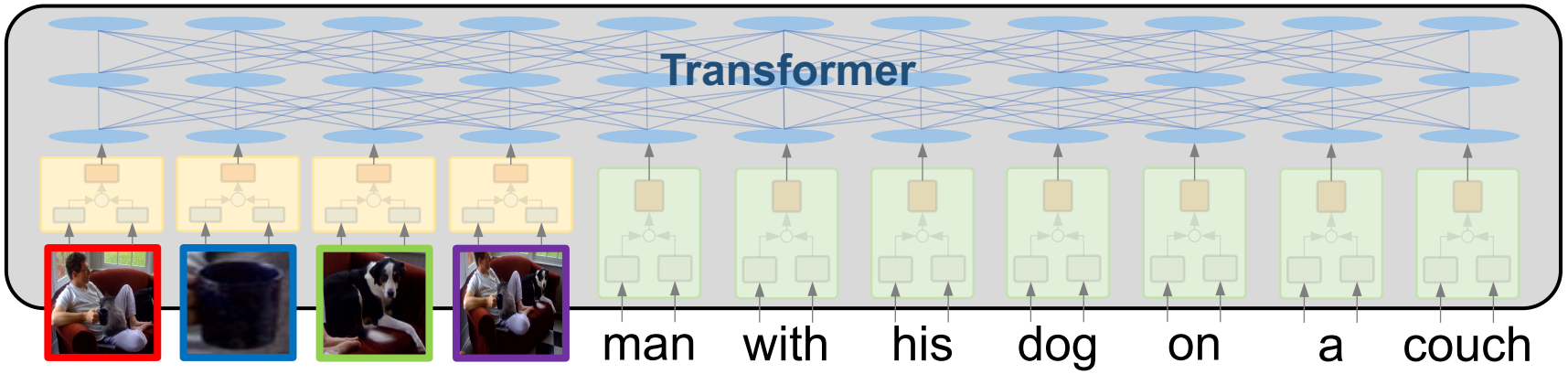
Pre-training Data

( , ‘man with his dog on a couch)

Visual and Language Features

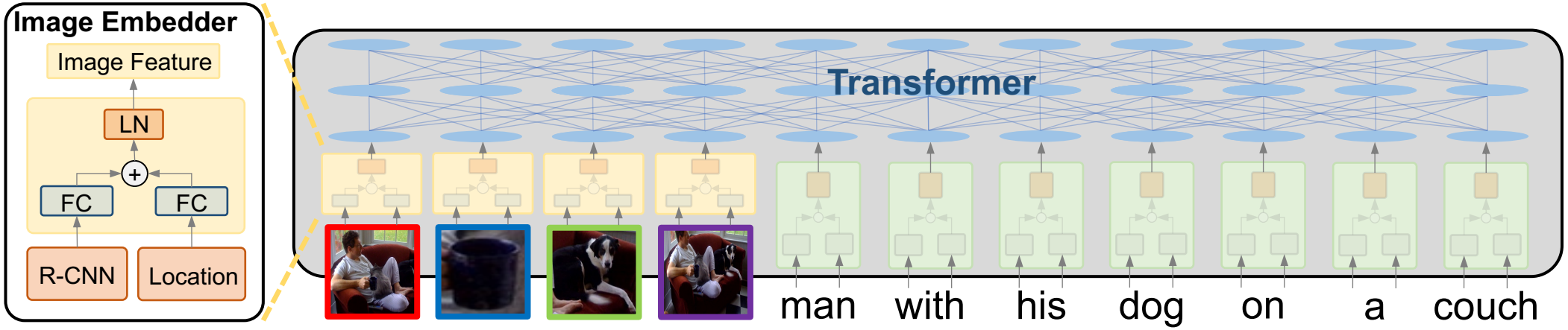
( , ‘man’ ‘with’ ‘his’ ‘dog’ ‘on’ ‘a’ ‘couch’)

Single-Stream Architecture



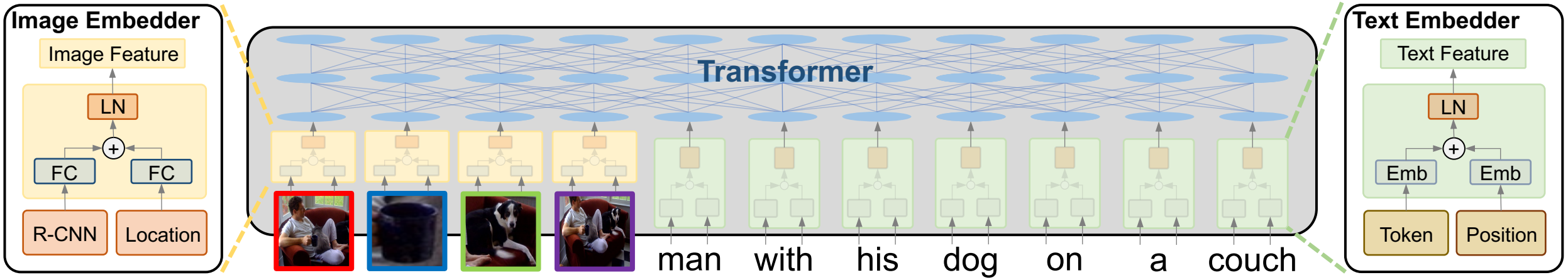
[UNITER; Chen et al2019]

Single-Stream Architecture



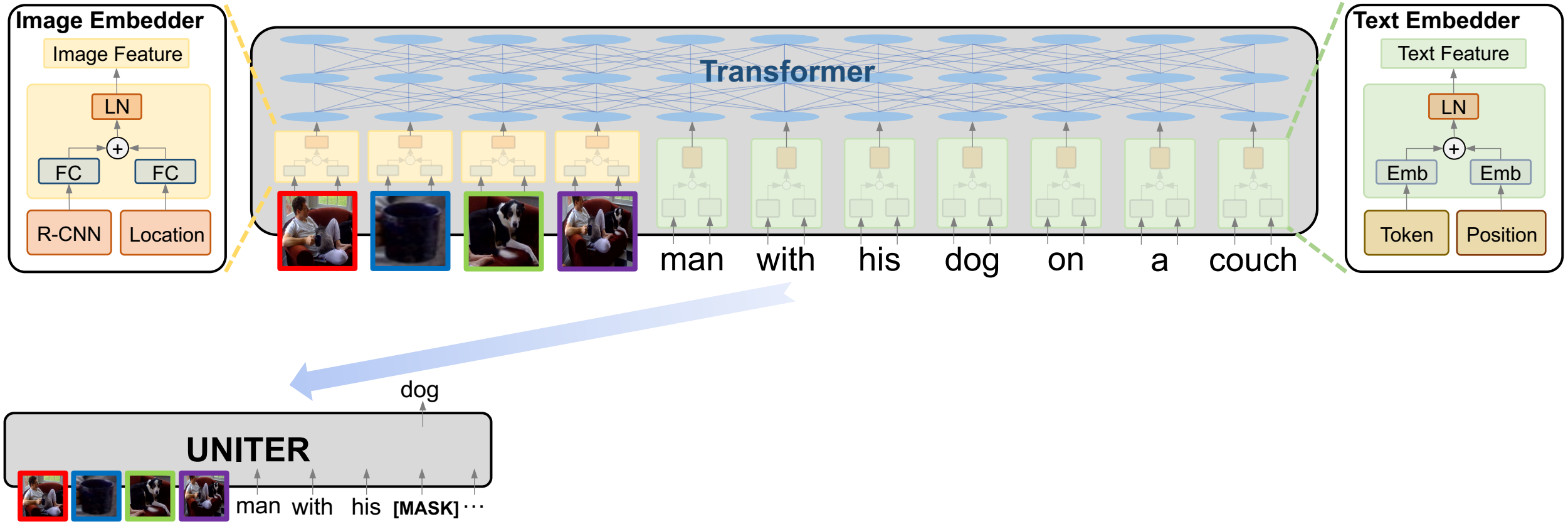
[UNITER; Chen et al2019]

Single-Stream Architecture



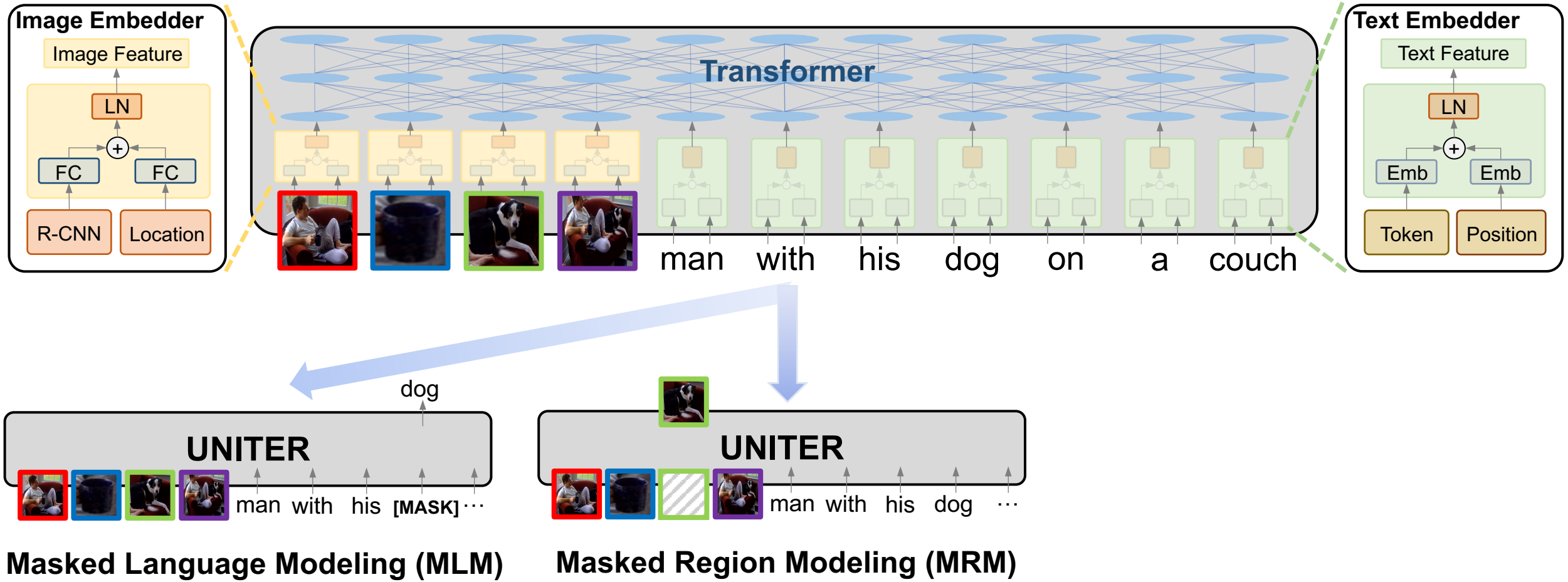
[UNITER; Chen et al2019]

Pre-training Tasks



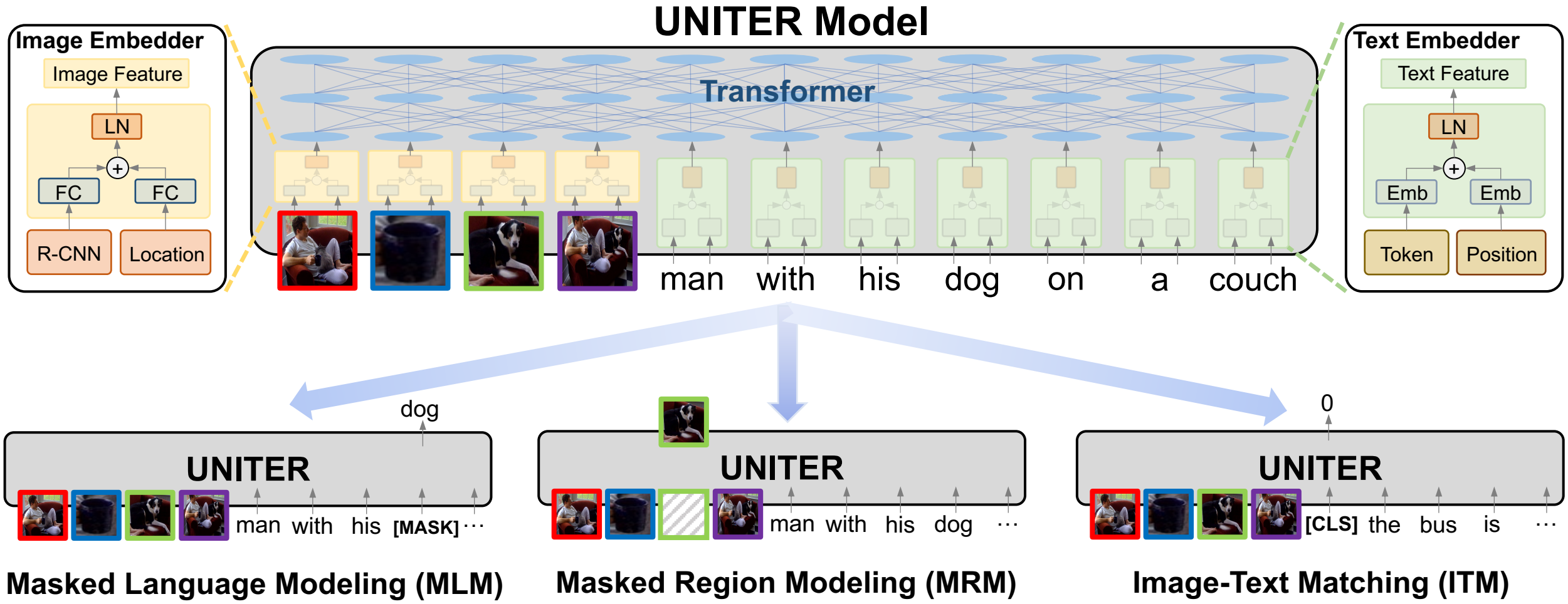
[UNITER; Chen et al2019]

Pre-training Tasks

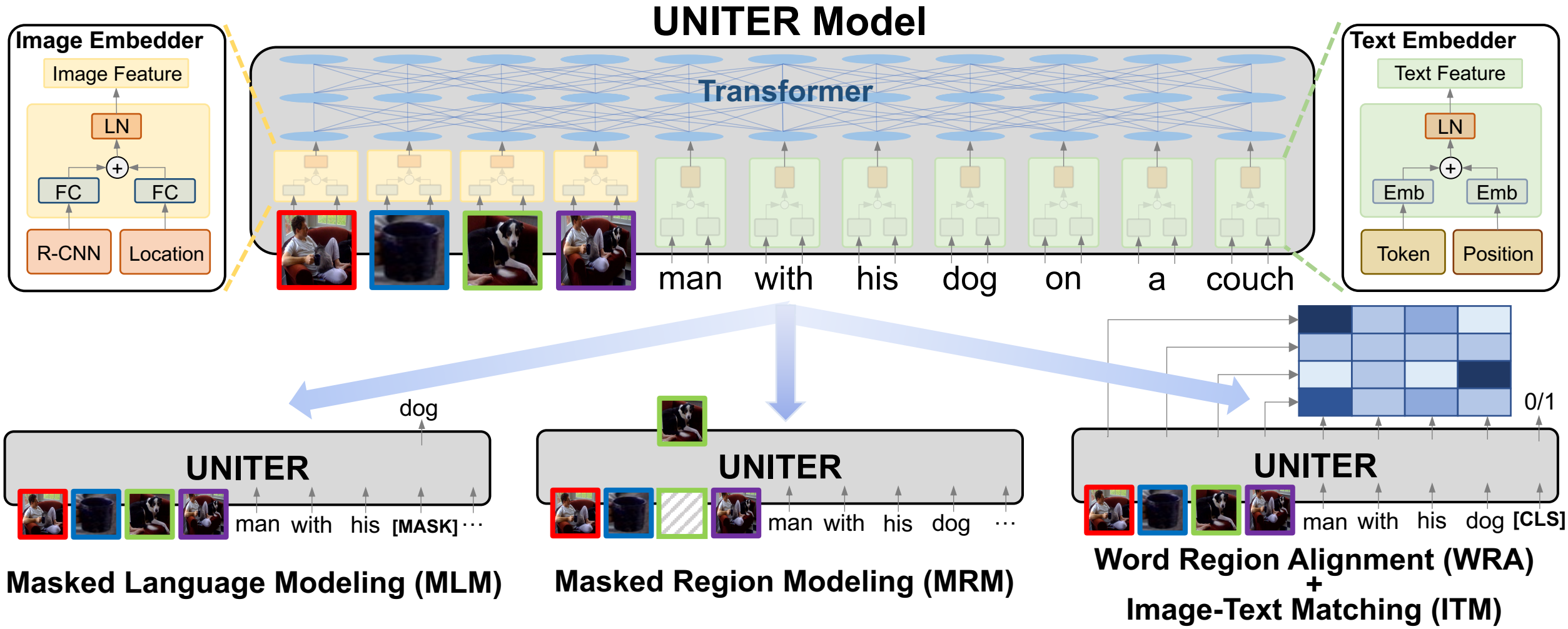


[UNITER; Chen et al2019]

Pre-training Tasks



Pre-training Tasks

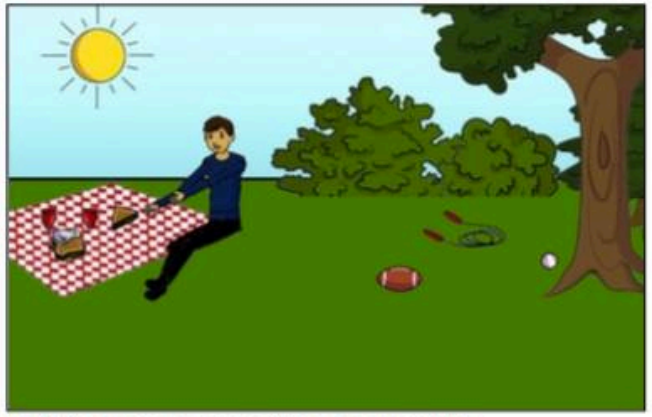


[UNITER; Chen et al2019]

Downstream Task 1: Visual Question Answering

What color are her eyes?
What is the mustache made of?

How many slices of pizza are there?
Is this a vegetarian pizza?

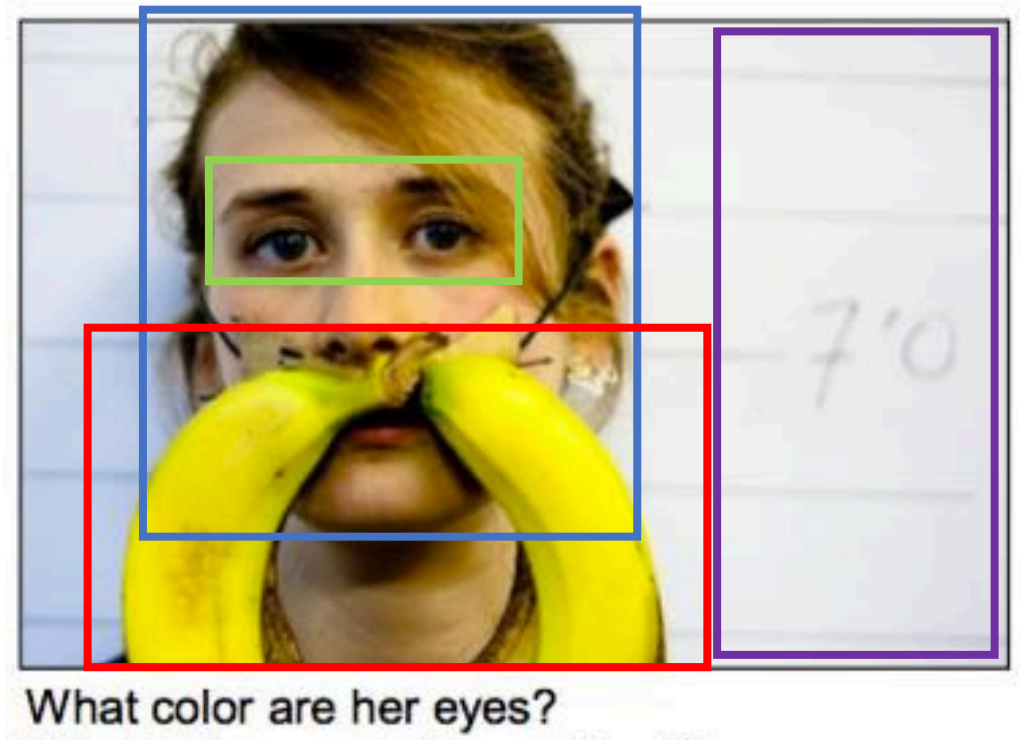


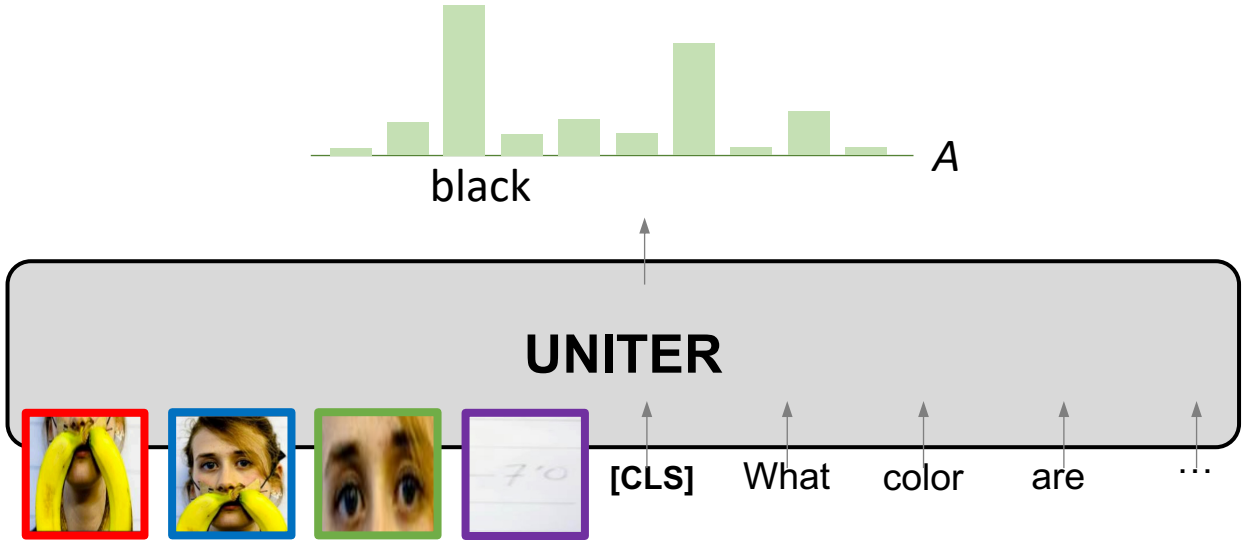
Is this person expecting company?
What is just under the tree?

Does it appear to be rainy?
Does this person have 20/20 vision?

[Antol et al., ICCV 2015]

Downstream Task 1: Visual Question Answering





Downstream Task 2: Visual Entailment

+

- *Two woman are holding packages.*
- *The sisters are hugging goodbye while holding to go packages after just eating lunch.*
- *The men are fighting outside a deli.*

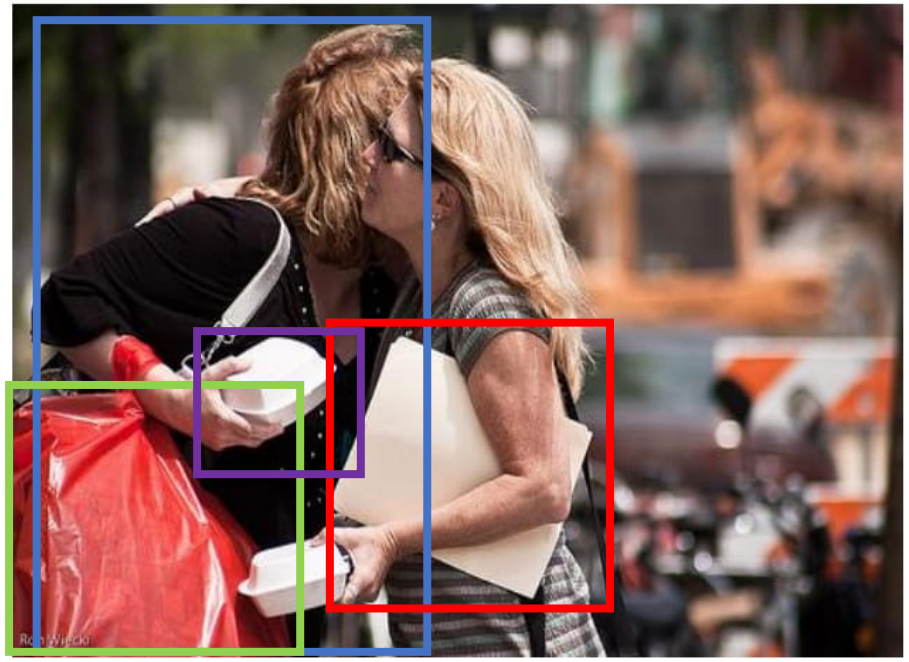
=

- *Entailment*
- *Neutral*
- *Contradiction*

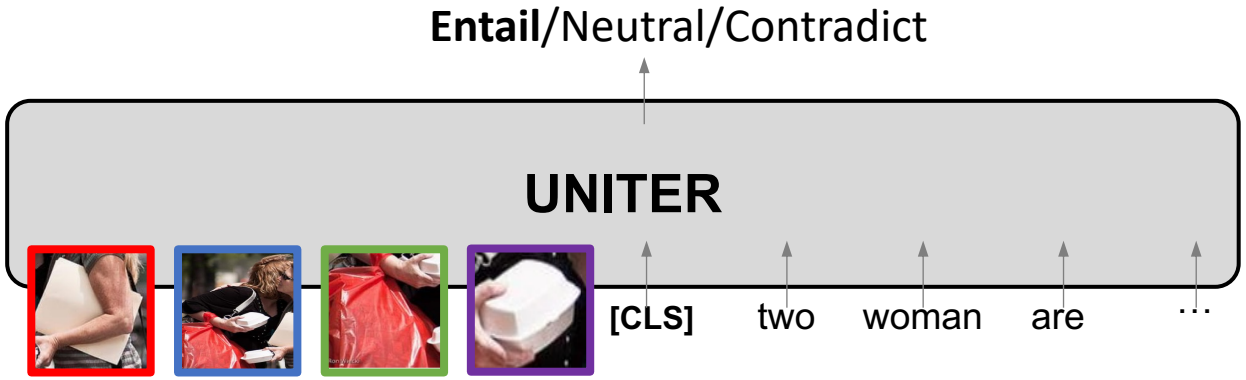
Premise *Hypothesis* *Answer*

[Xie et al., 2019]

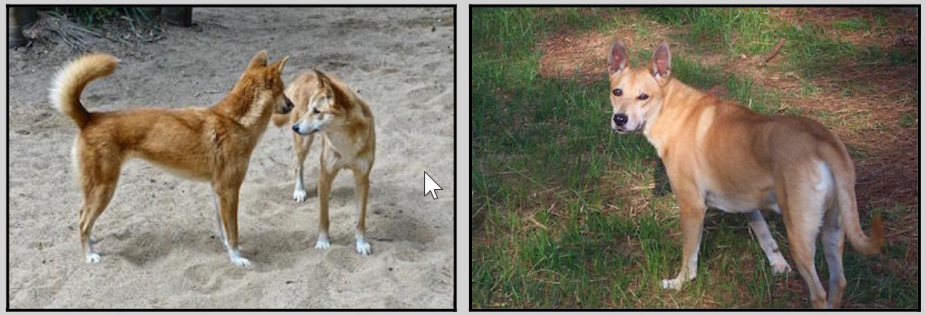
Downstream Task 2: Visual Entailment



Two woman are holding packages.



Downstream Task 3: Natural Language for Visual Reasoning



The left image contains twice the number of dogs as the right image, and at least two dogs in total are standing.

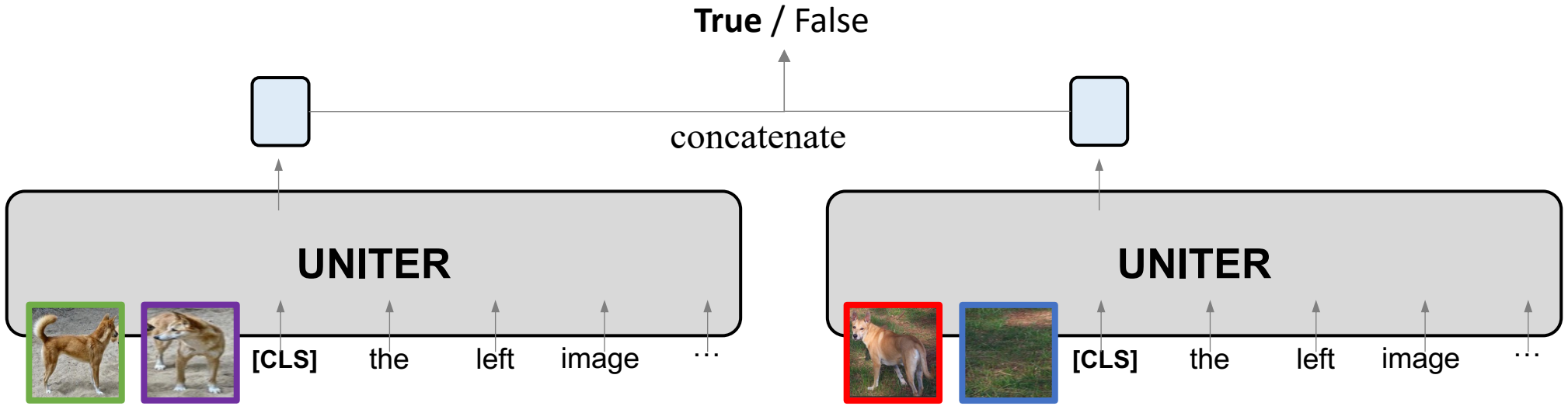
true

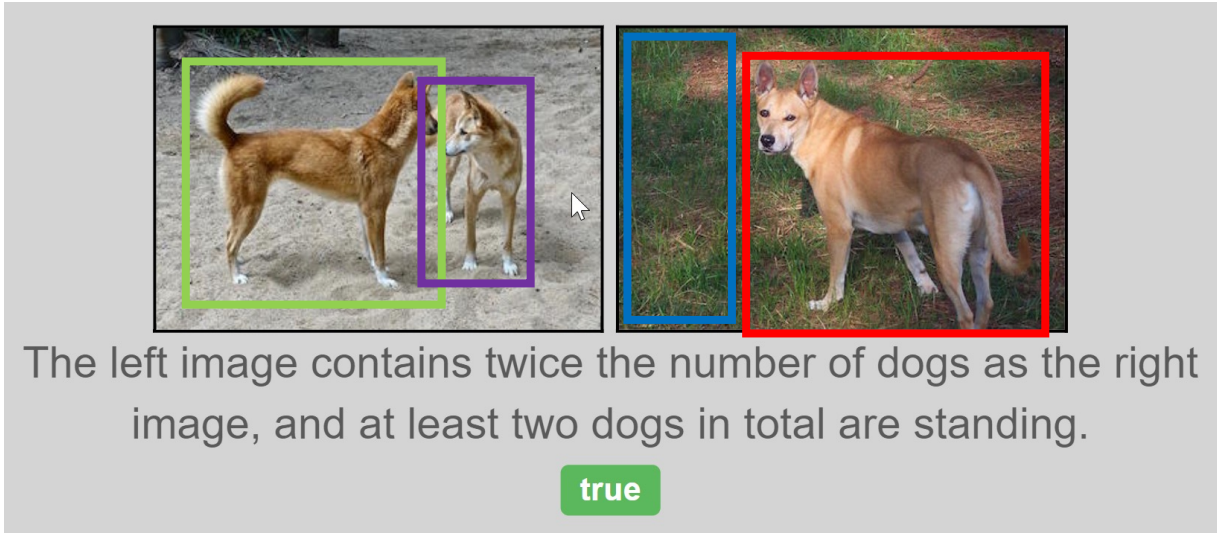
One image shows exactly two brown acorns in back-to-back caps on green foliage.

false

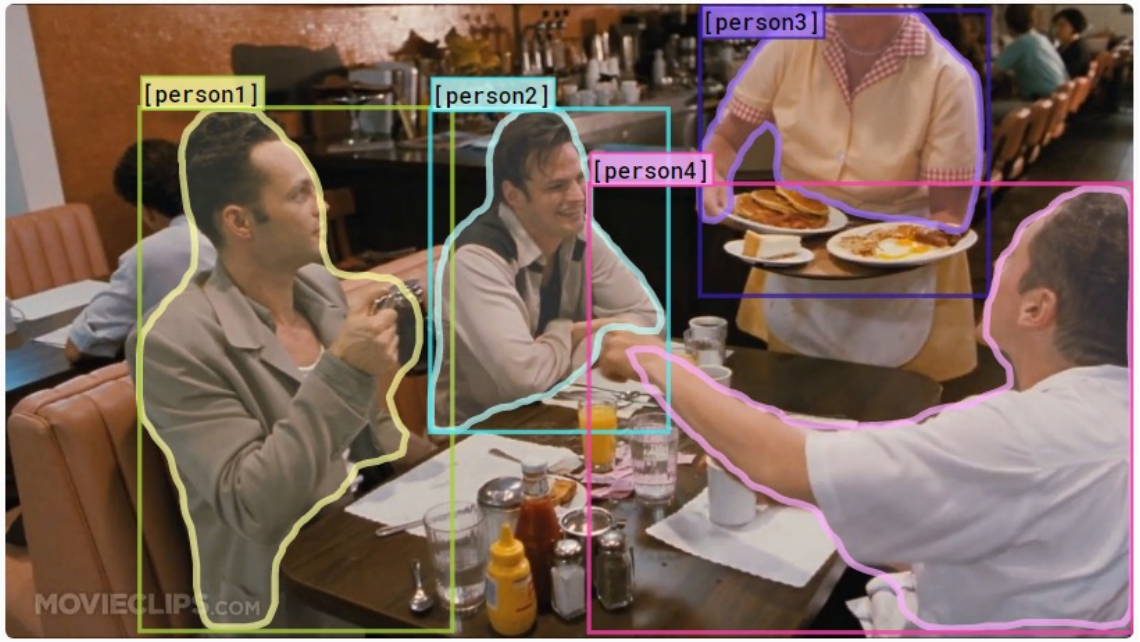
[Suhr et al., ACL 2019]

Downstream Task 3: Natural Language for Visual Reasoning





Downstream Task 4: Visual Commonsense Reasoning



Why is [person4] pointing at [person1]?

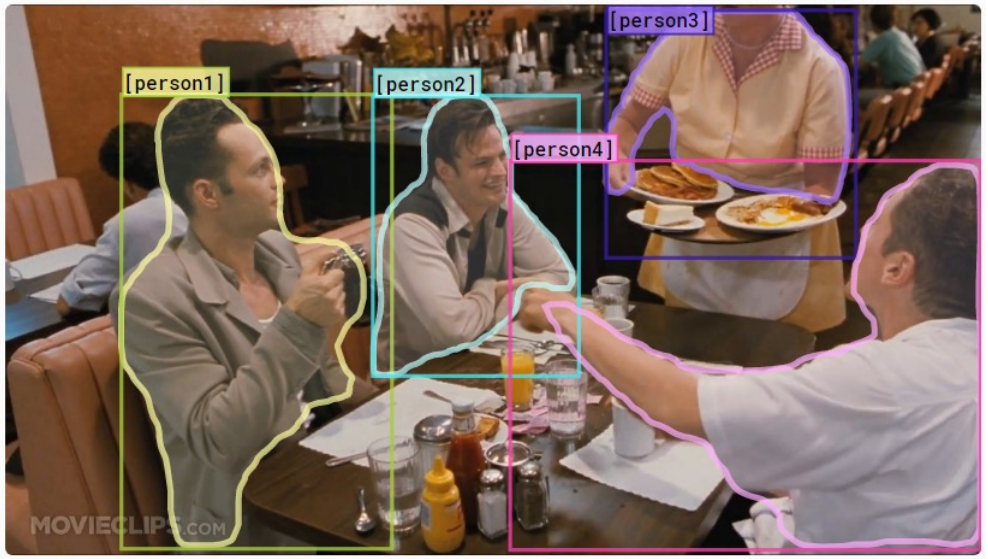
a) He is telling [person3] that [person1] ordered the pancakes.
b) He just told a joke.
c) He is feeling accusatory towards [person1].
d) He is giving [person1] directions.

I choose (a) because:

a) [person1] has the pancakes in front of him.
b) [person4] is taking everyone's order and asked for clarification.
c) [person3] is looking at the pancakes and both she and [person2] are smiling slightly.
d) [person3] is delivering food to the table, and she might not know whose order is whose.

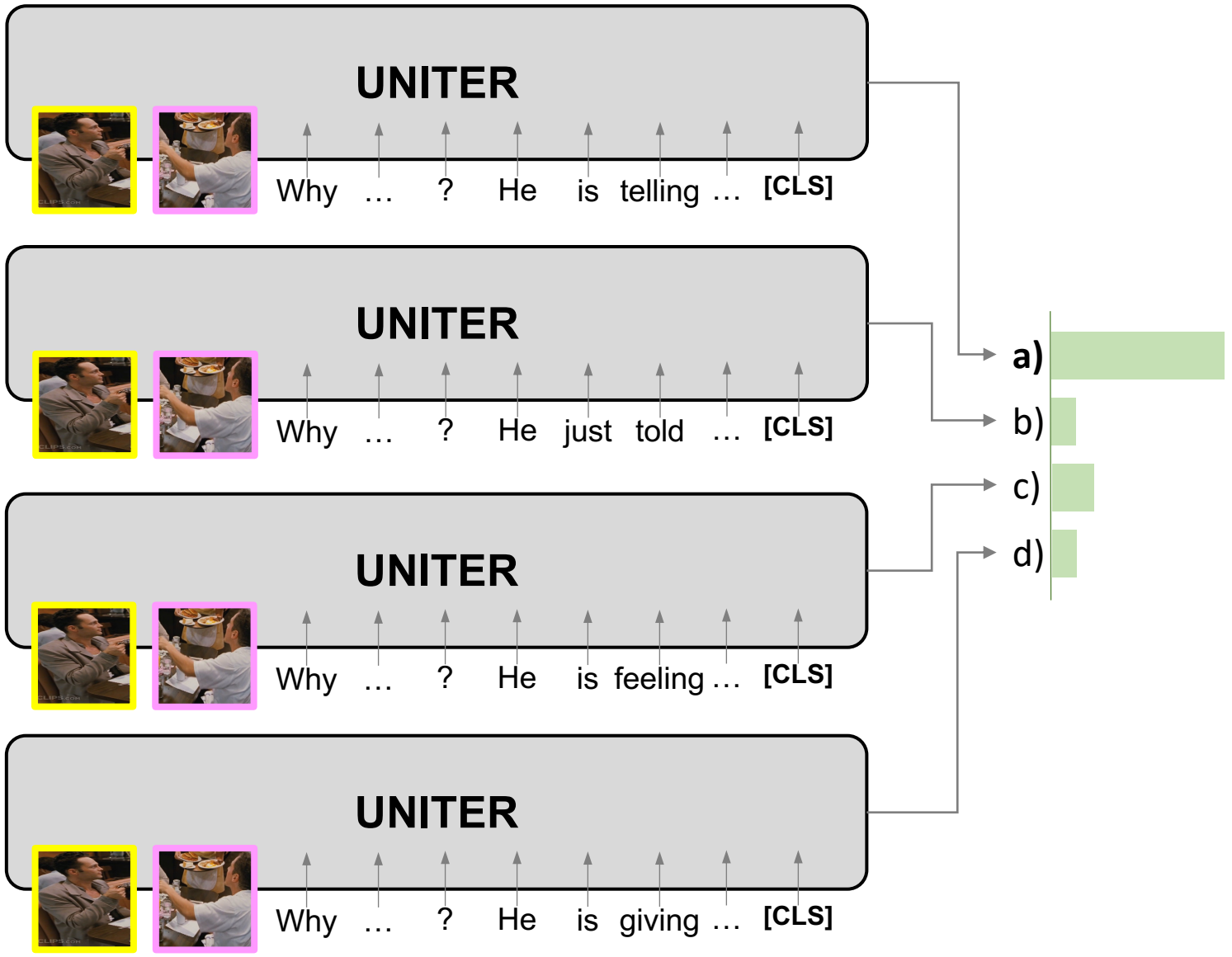
[Zellers et al., CVPR 2019]

Downstream Task 4: Visual Commonsense Reasoning



Why is [person4] pointing at [person1]?

a) He is telling [person3] that [person1] ordered the pancakes.
b) He just told a joke.
c) He is feeling accusatory towards [person1].
d) He is giving [person1] directions.

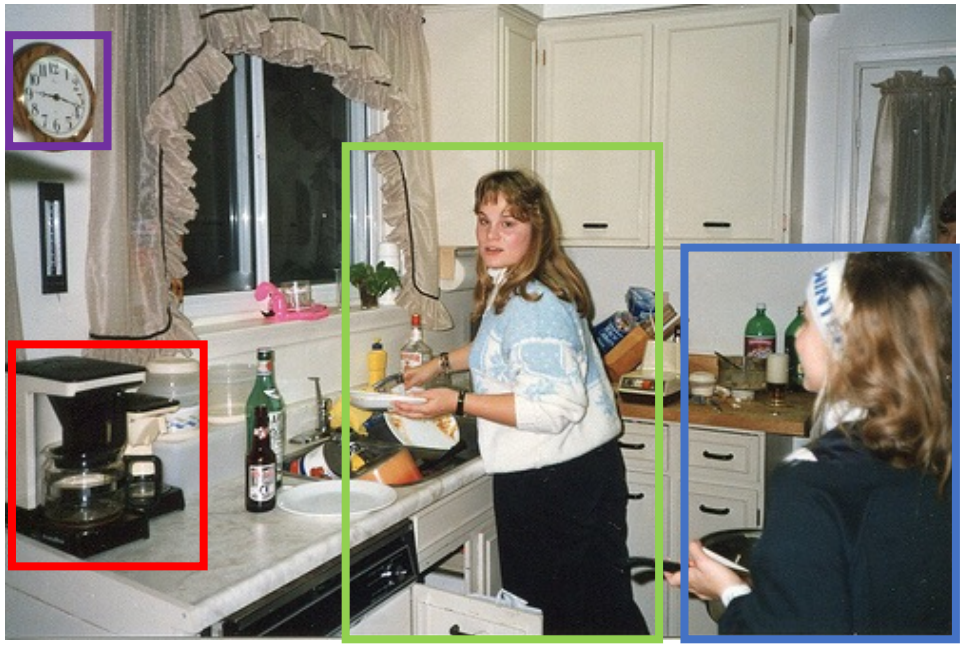


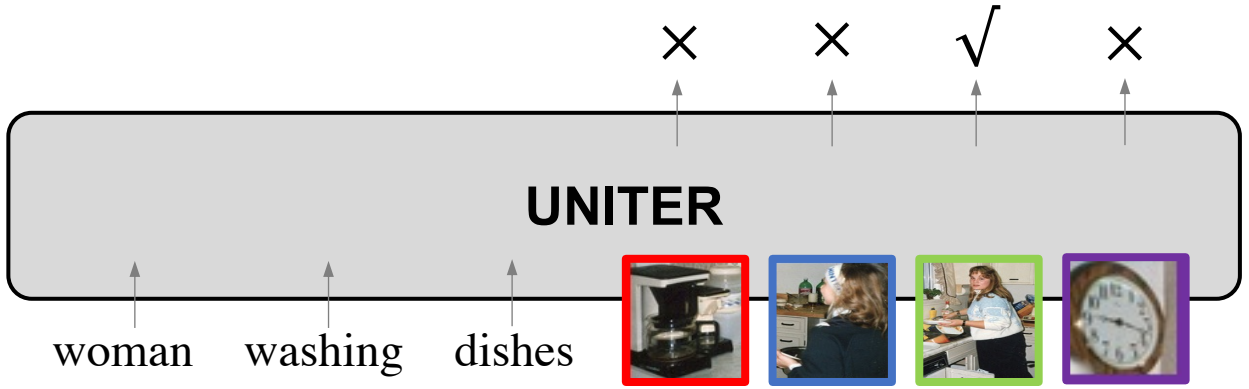
Downstream Task 5: Referring Expression Comprehension

woman washing dishes

[Kazemzadeh et al., EMNLP 2014]

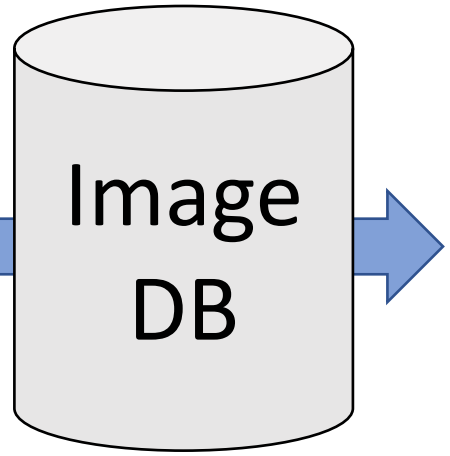
Downstream Task 5: Referring Expression Comprehension





Downstream Task 6: Image-Text Retrieval

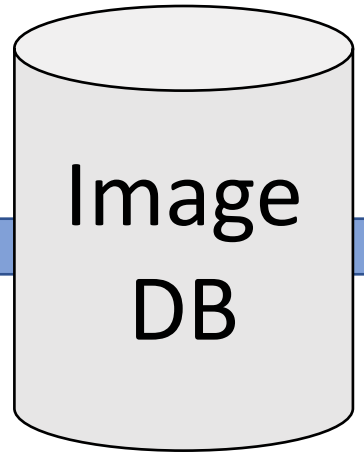
“a girl with a cat on grass”



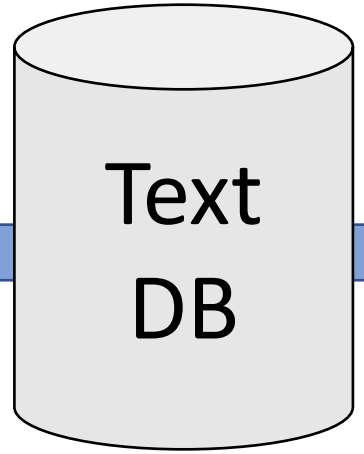
...

Downstream Task 6: Image-Text Retrieval

"a girl with a cat on grass"



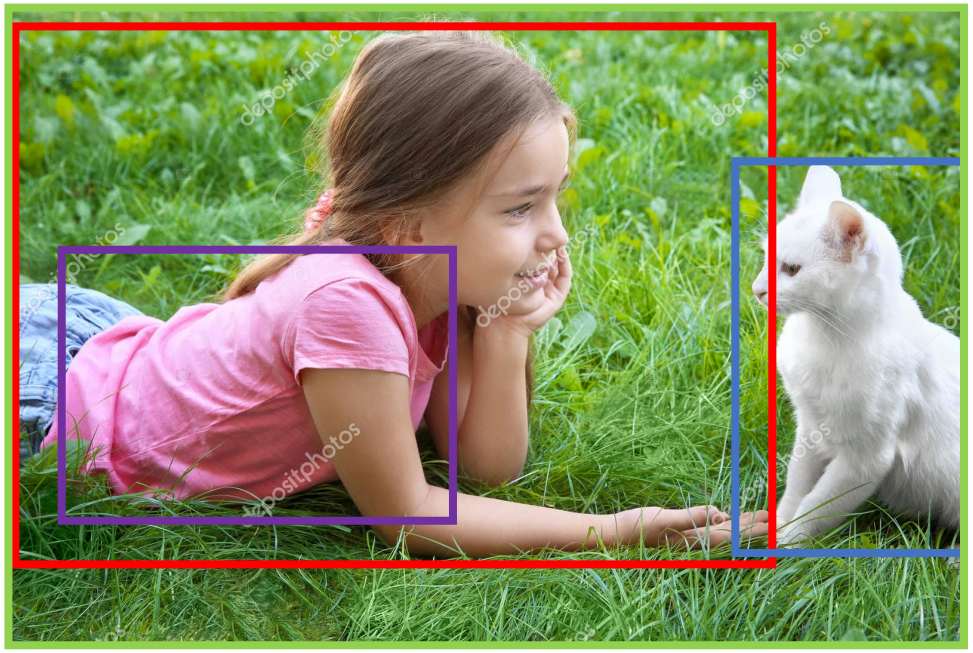
...

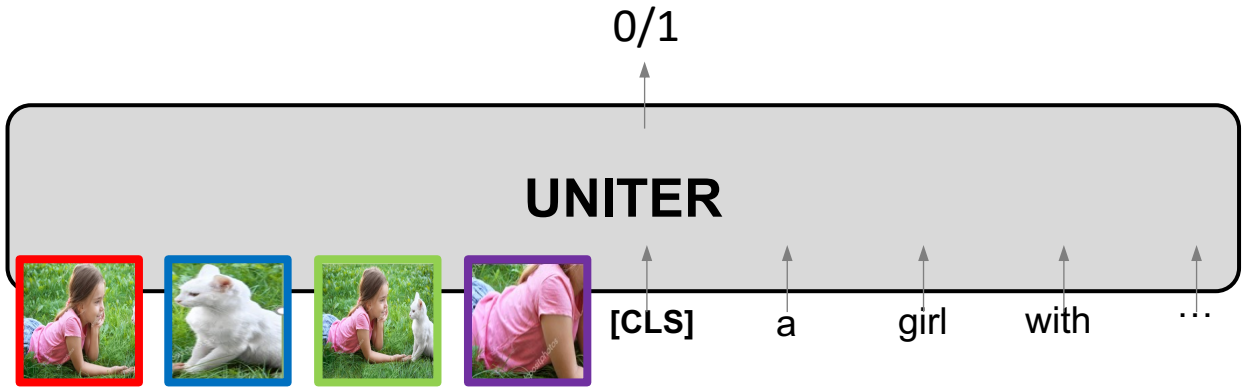


"four people with ski poles in their hands in the snow"
"four skiers hold on to their poles in a snowy forest"
"a group of young men riding skis"
"skiers pose for a picture while outside in the woods"
"a group of people cross country skiing in the woods"

⋮

Downstream Task 6: Image-Text Retrieval

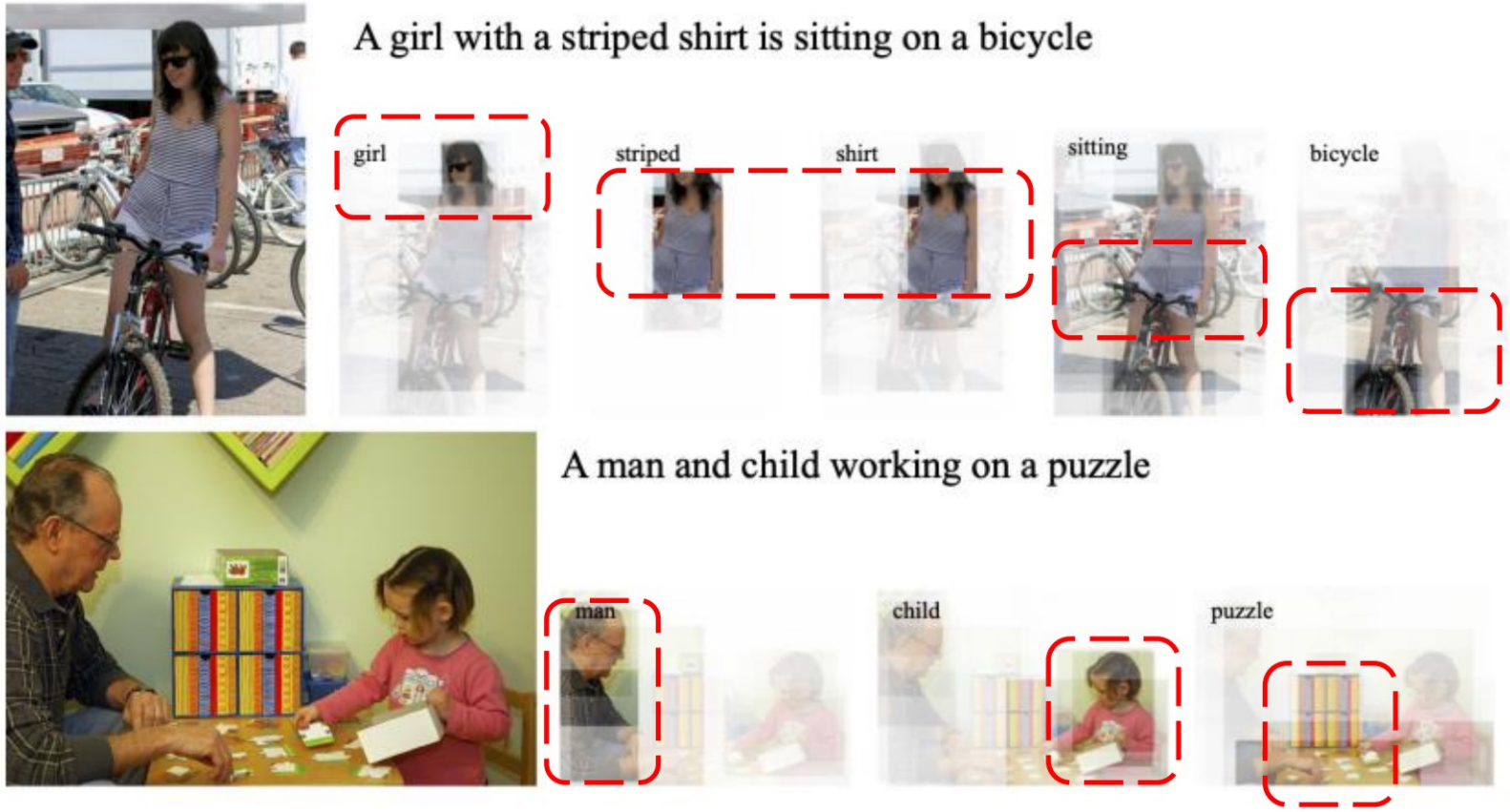




Lee et al., ECCV 2018

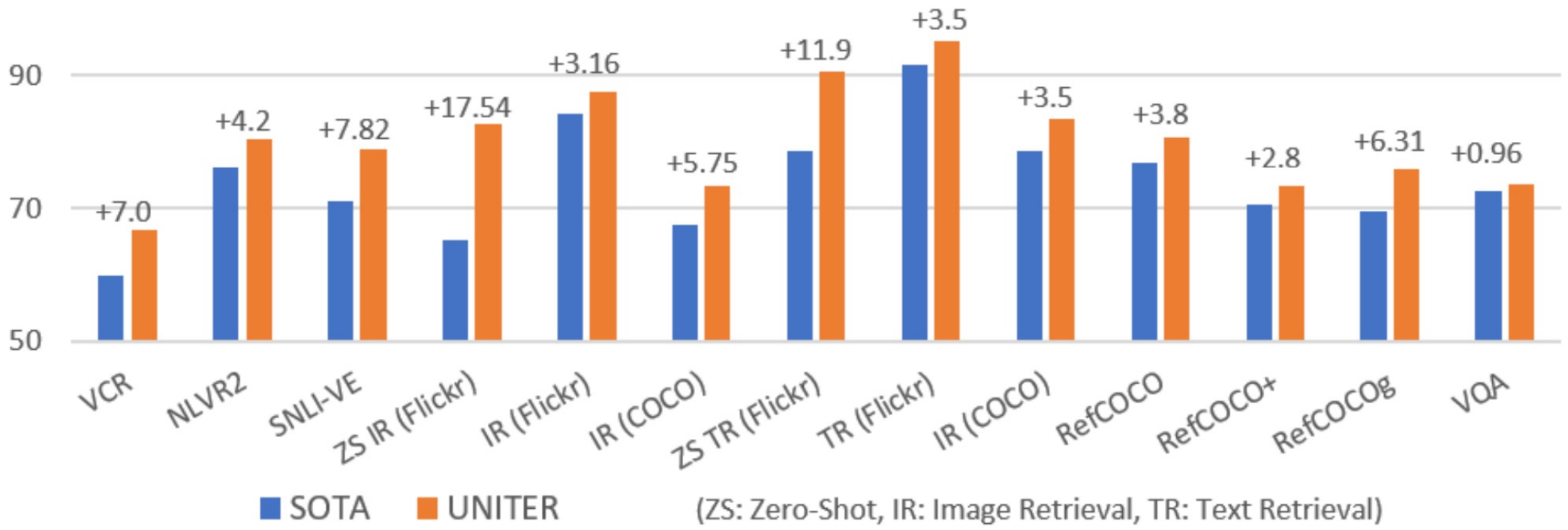
Visualization (Text-to-Image Attention)

- UNITER learns local cross-modality alignment between regions and tokens



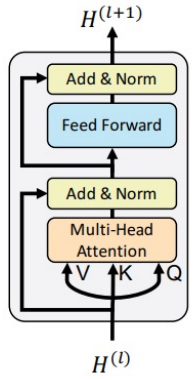
State-of-the-Art Results

- UNITER outperformed both task-specific and pre-trained SOTA models over nine V+L tasks (as of Sep 2019 until early 2020)

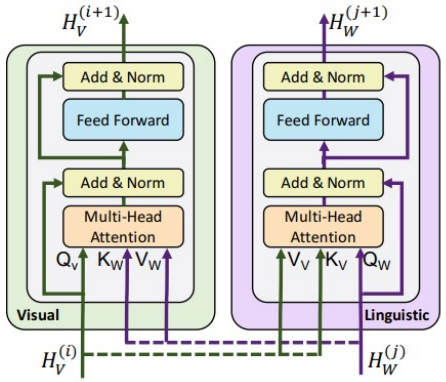


Two-stream Models

- Two transformers are used to encode two modalities independently first
- A third transformer is stacked on top for multi-modal fusion



(a) Standard encoder transformer block



(b) Our co-attention transformer layer

LXMERT, EMNLP 2019

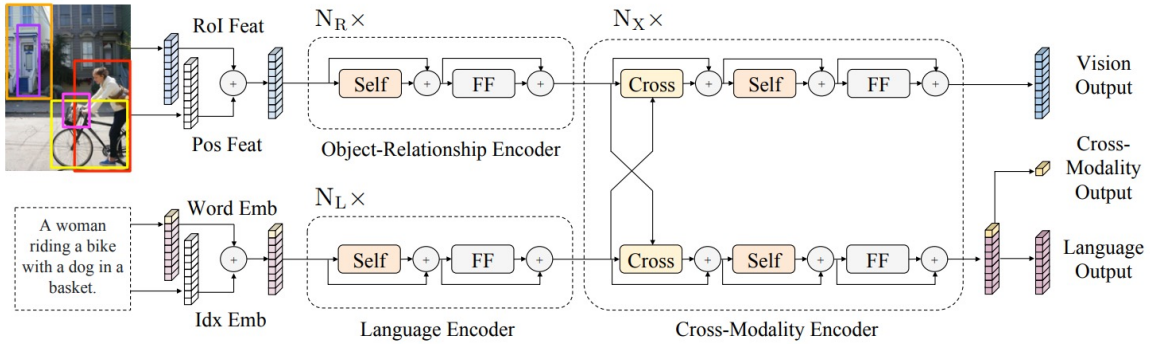
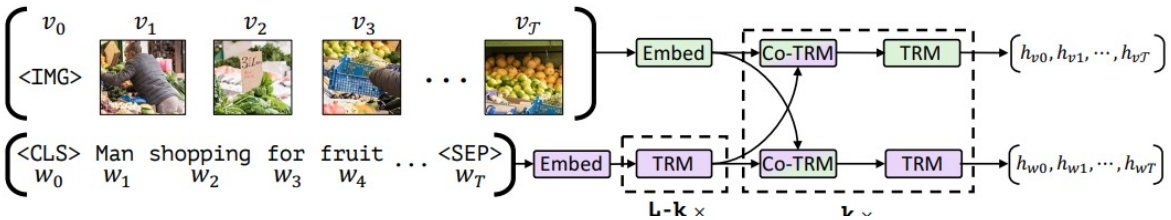


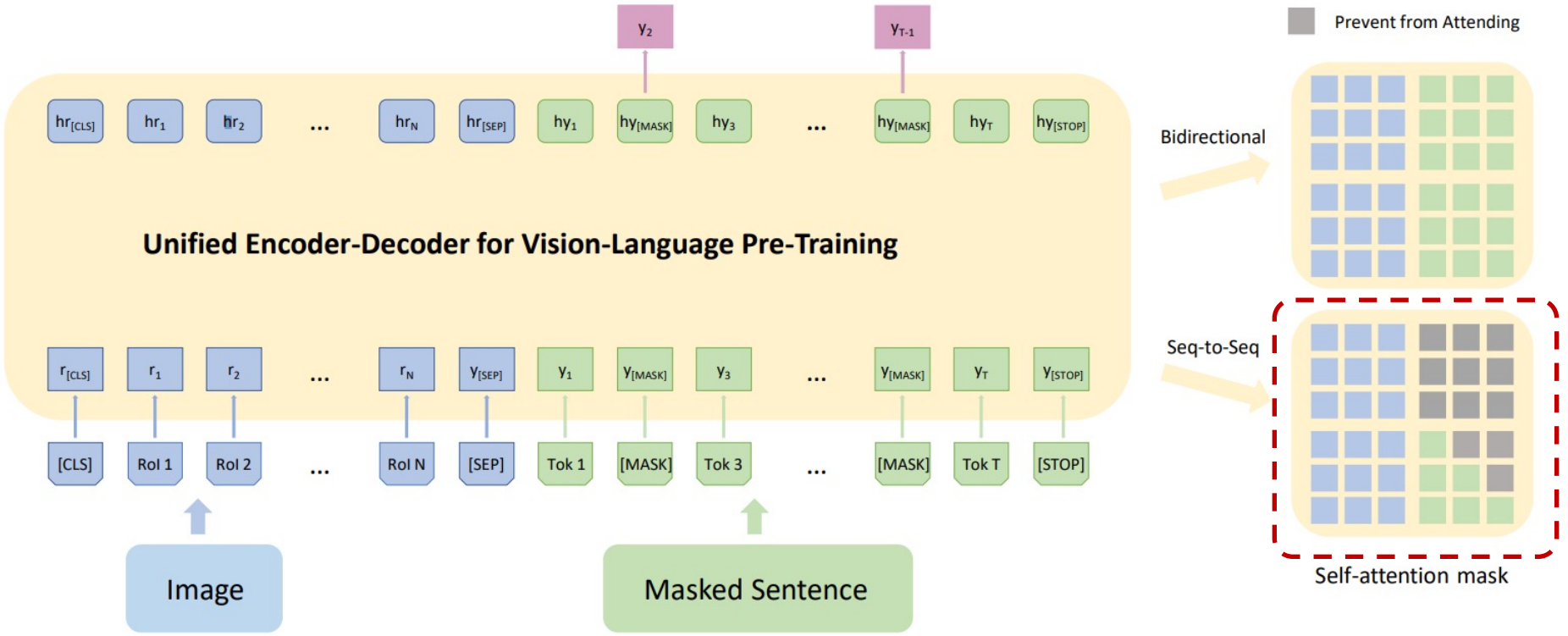
Figure 1: The LXMERT model for learning vision-and-language cross-modality representations. 'Self' and 'Cross' are abbreviations for self-attention sub-layers and cross-attention sub-layers, respectively. 'FF' denotes a feed-forward sub-layer.

ViLBERT, NeurIPS 2019



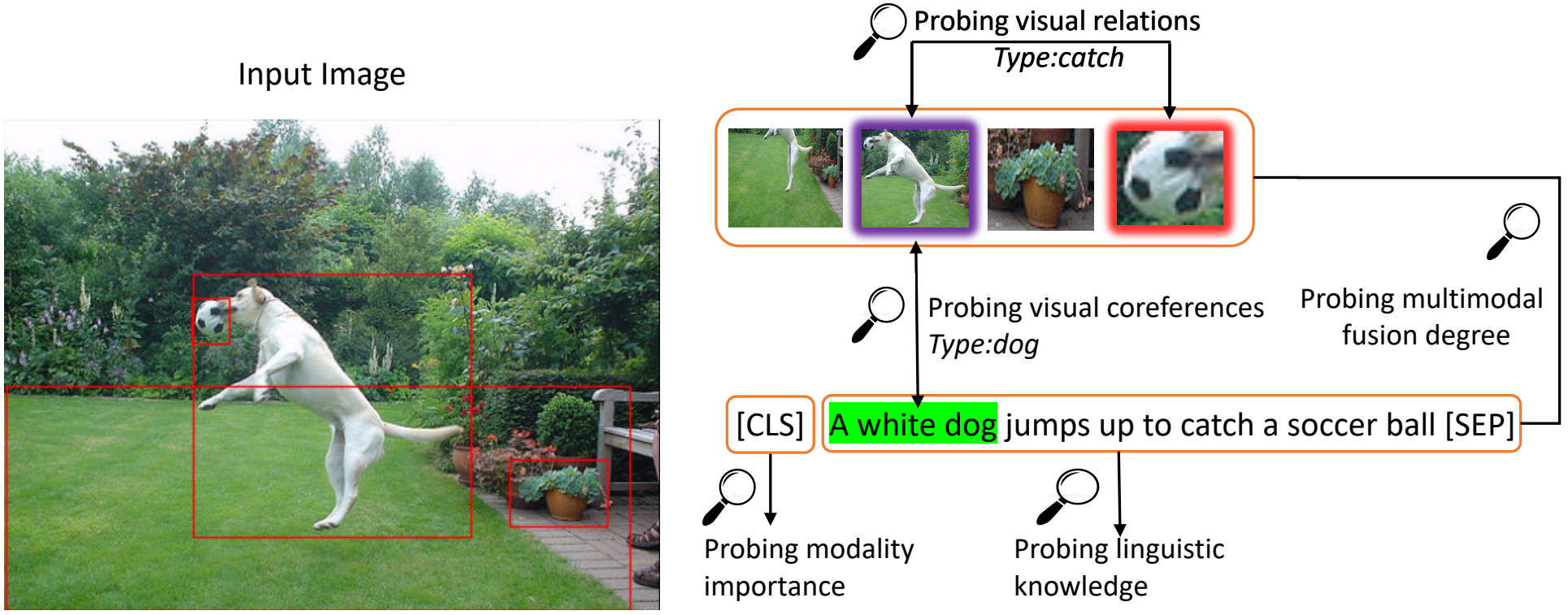
VLP for image captioning

- An additional seq2seq attention mask is designed



[1 Unified Vision-Language Pre-Training for Image Captioning and VQA, AAAI 2020

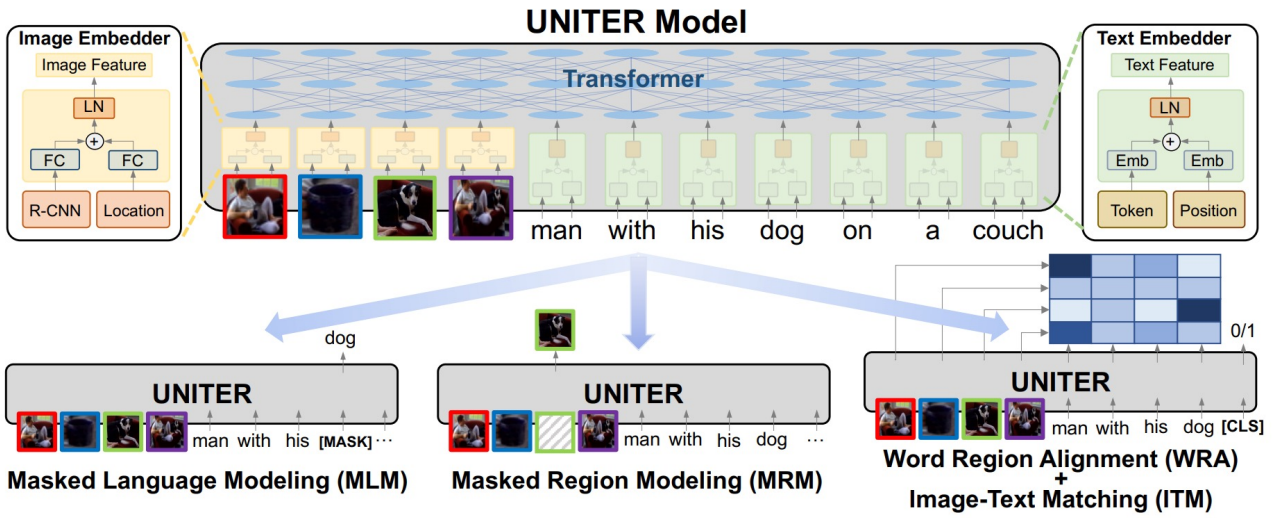
What do VLP Models Learn?



[1 Behind the Scene: Revealing the Secrets of Pre-trained Vision-and-Language Models, ECCV 2020

Take UNITER As an Example

- Attention weight probing
  - 12 layers x 12 heads = 144 attention weight matrices
- Embedding probing
  - 768-dim x 12 layers

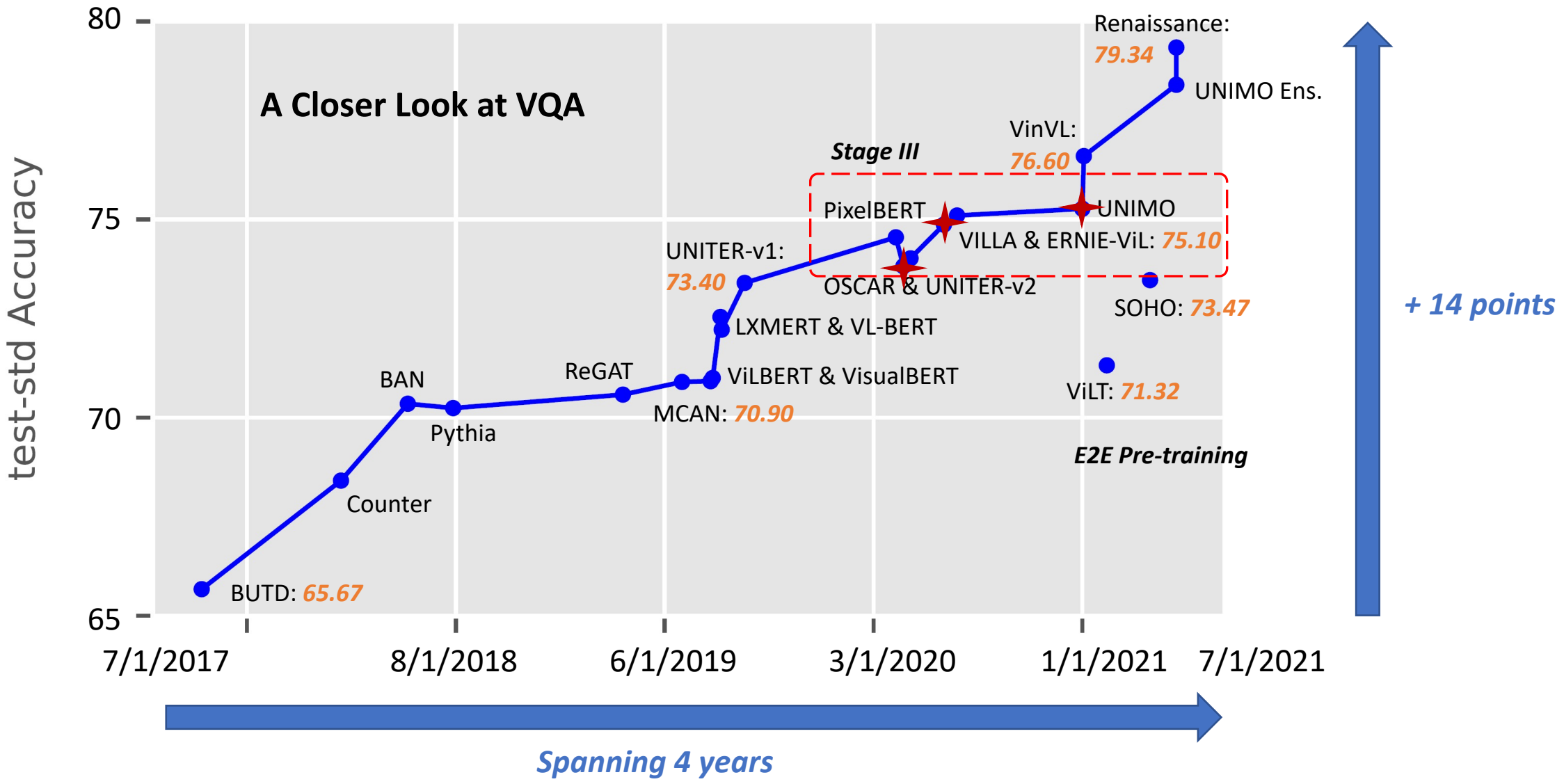


[1 Behind the Scene: Revealing the Secrets of Pre-trained Vision-and-Language Models, ECCV 2020

Take-home Message

- Probe multimodal fusion degree:
  - *Deep to Profound*: Deeper layers lead to more intertwined multimodal fusion
- Probe modality importance:
  - *Who Pulls More Strings*: Textual modality is more dominant than image
- Probe visual coreferences/visual relations/linguistic knowledge:
  - *Winner Takes All*: A subset of heads is specialized for cross-modal interaction
  - *Secret Liaison Revealed*: Cross-modality fusion registers visual relations
  - *No Lost in Translation*: Pre-trained V+L models encode rich linguistic knowledge

[1 Behind the Scene: Revealing the Secrets of Pre-trained Vision-and-Language Models, ECCV 2020

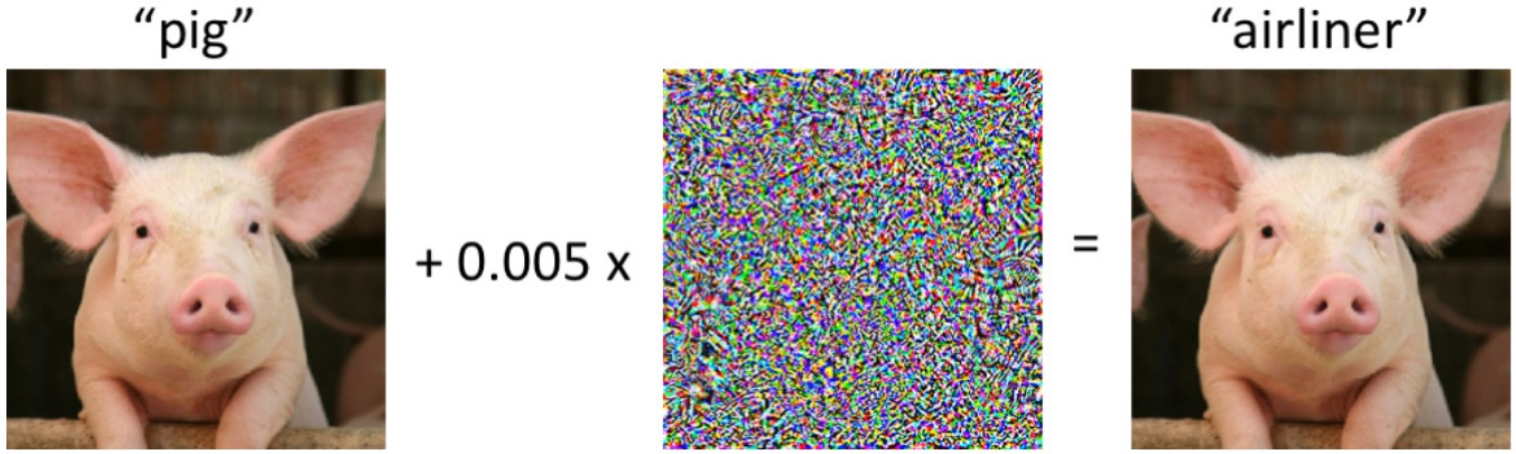

A Closer Look at VQA
test-std Accuracy
80
75
70
65
7/1/2017
8/1/2018
6/1/2019
3/1/2020
1/1/2021
7/1/2021
BUTD: 65.67
Counter
BAN
Pythia
ReGAT
MCAN: 70.90
ViLBERT & VisualBERT
LXMERT & VL-BERT
UNITER-v1:
73.40
Stage III
PixelBERT
OSCAR & UNITER-v2
VinVL:
76.60
UNIMO
VILLA & ERNIE-ViL: 75.10
Renaissance:
79.34
UNIMO Ens.
SOHO: 73.47
ViLT: 71.32
E2E Pre-training
+ 14 points
Spanning 4 years

VILLA: Vision-and-Language Large-scale Adversarial Training

Preliminary: What's Adversarial Attack?

- Neural Networks are prone to label-preserving adversarial examples

Computer Vision:



Natural Language Processing:

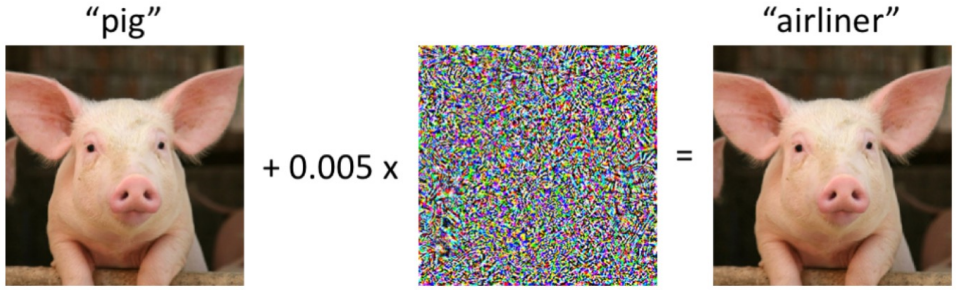
Original: What is the oncorhynchus also called? **A:** chum salmon	**Original:** How long is the Rhine? **A:** 1,230 km
Changed: What's the oncorhynchus also called? **A:** keta	**Changed:** How long is the Rhine?? **A:** more than 1,050,000
(b) Example for (*WP is→WP's*)	(c) Example for (*?→??*)

[1] Explaining and harnessing adversarial examples. *arXiv:1412.6572*
[2] Semantically equivalent adversarial rules for debugging nlp models. *ACL (2018)*

Preliminary: What's Adversarial Training (AT)?

- A min-max game to harness adversarial examples

$$\min_{\theta} \mathop{\mathbb{E}}_{(x,y)\sim\widehat{\mathcal{D}}} \left[\max_{\delta \in S} \mathcal{L}(x+\delta, y; \theta) \right]$$

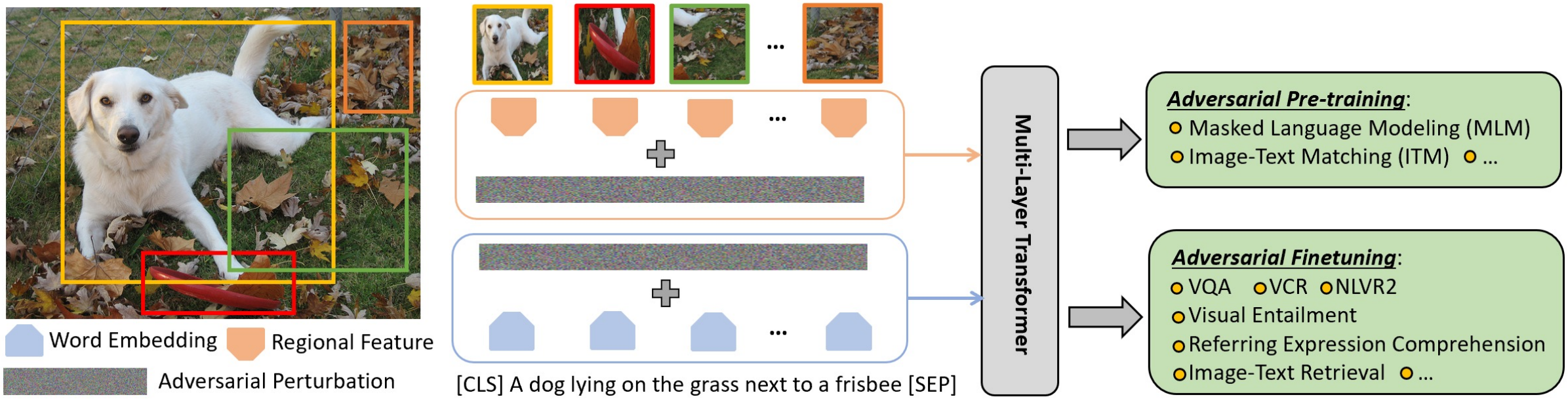


- Use adversarial examples as additional training samples
  - On one hand, we try to find perturbations that maximize the empirical risk
  - On the other hand, the model tries to make correct predictions on adversarial examples
- *What doesn't kill you makes you stronger!*

Explaining and harnessing adversarial examples. *arXiv:1412.6572*

What's the Recipe of VILLA?

- Ingredient #1: Adversarial pre-training + finetuning
- Ingredient #2: Perturbations in the embedding space
- Ingredient #3: Enhanced adversarial training algorithm



Enhanced AT Algorithm

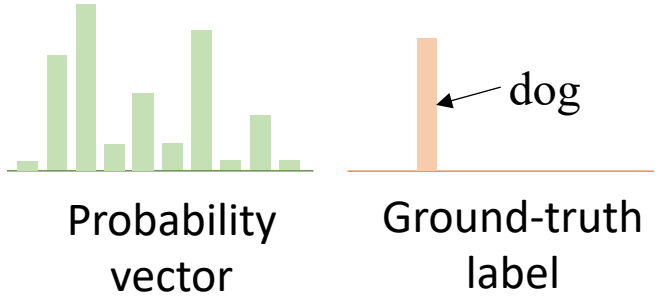
- Training objective:

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt}, \boldsymbol{y}) \sim \mathcal{D}} \Big[\mathcal{L}_{std}(\boldsymbol{\theta}) + \mathcal{R}_{at}(\boldsymbol{\theta}) + \alpha \cdot \mathcal{R}_{kl}(\boldsymbol{\theta}) \Big]$$

- Cross-entropy loss on clean data:

$$\mathcal{L}_{std}(\boldsymbol{\theta}) = L(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt}), \boldsymbol{y})$$

, A [MASK] lying on the grass next to a frisbee



Probability vector

Ground-truth label

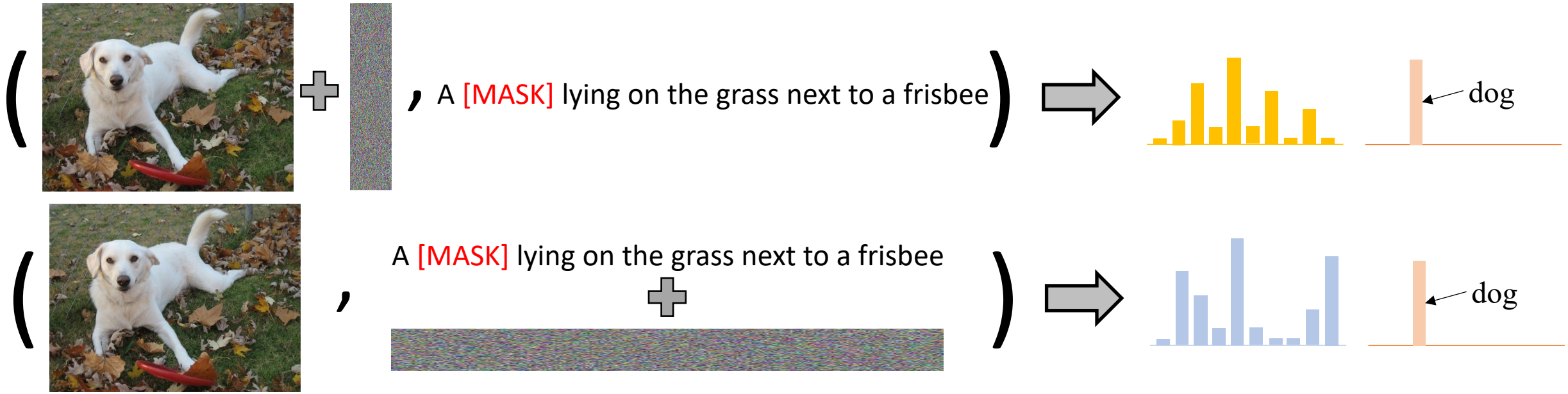
Enhanced AT Algorithm

- Training objective:

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt}, \boldsymbol{y}) \sim \mathcal{D}} \Big[\mathcal{L}_{std}(\boldsymbol{\theta}) + \mathcal{R}_{at}(\boldsymbol{\theta}) + \alpha \cdot \mathcal{R}_{kl}(\boldsymbol{\theta}) \Big]$$

- Cross-entropy loss on adversarial embeddings:

$$\mathcal{R}_{at}(\boldsymbol{\theta}) = \max_{||\boldsymbol{\delta}_{img}|| \leq \epsilon} L(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img} + \boldsymbol{\delta}_{img}, \boldsymbol{x}_{txt}), \boldsymbol{y}) + \max_{||\boldsymbol{\delta}_{txt}|| \leq \epsilon} L(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt} + \boldsymbol{\delta}_{txt}), \boldsymbol{y})$$



Enhanced AT Algorithm

- Training objective:

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt}, \boldsymbol{y}) \sim \mathcal{D}} \Big[\mathcal{L}_{std}(\boldsymbol{\theta}) + \mathcal{R}_{at}(\boldsymbol{\theta}) + \alpha \cdot \mathcal{R}_{kl}(\boldsymbol{\theta}) \Big]$$

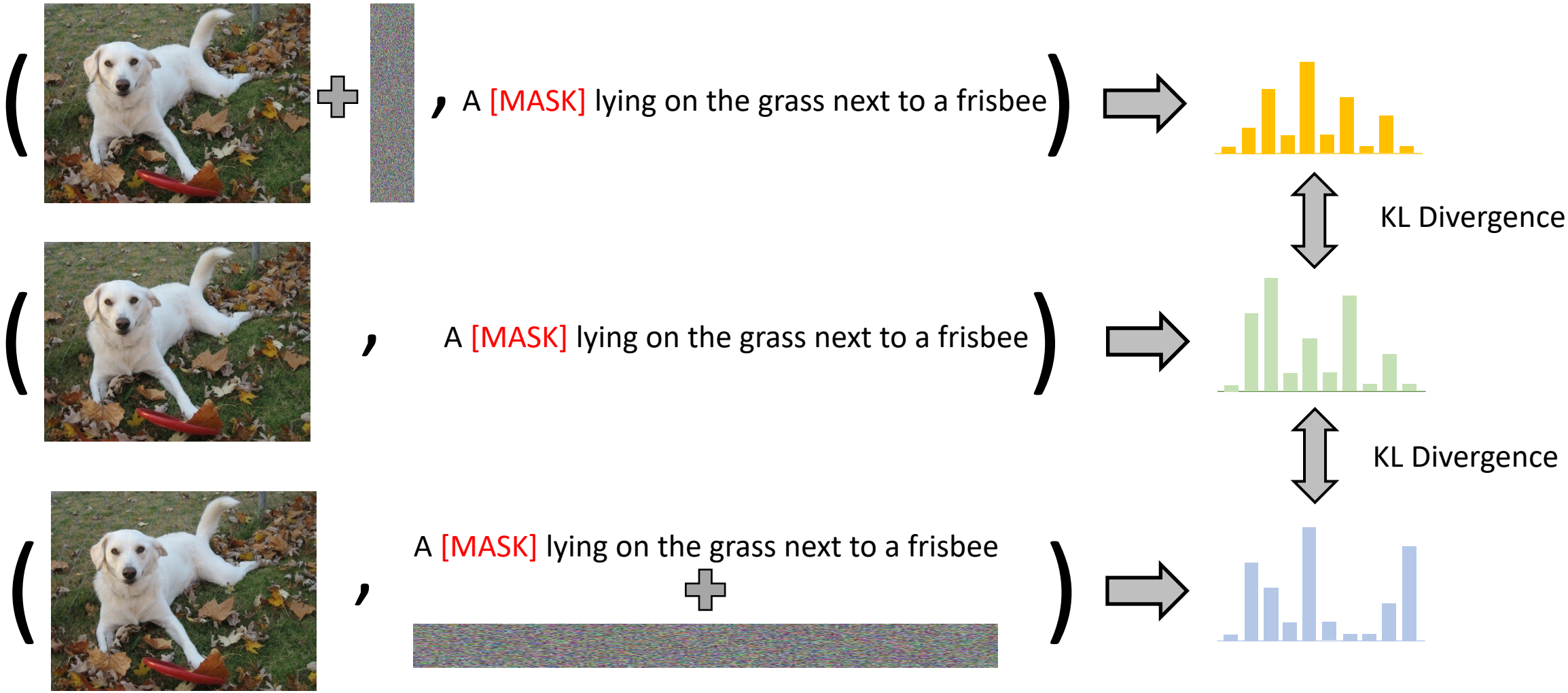
- KL-divergence loss for fine-grained adversarial regularization

$$\begin{aligned} \mathcal{R}_{kl}(\boldsymbol{\theta}) = & \max_{||\boldsymbol{\delta}_{img}|| \leq \epsilon} L_{kl}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img} + \boldsymbol{\delta}_{img}, \boldsymbol{x}_{txt}), f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt})) \\ & + \max_{||\boldsymbol{\delta}_{txt}|| \leq \epsilon} L_{kl}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt} + \boldsymbol{\delta}_{txt}), f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt})) \,, \end{aligned}$$

where $L_{kl}(p, q) = \mathrm{KL}(p||q) + \mathrm{KL}(q||p)$.

- Not only label-preserving, but the confidence level of the prediction between clean data and adversarial examples should also be close

Enhanced AT Algorithm



Training Algorithm

Enable AT for large-scale training and promote diverse adversaries

Algorithm 1 "Free" Multi-modal Adversarial Training used in VILLA.

Require: Training samples $\mathcal{D} = \{(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt}, \boldsymbol{y})\}$, perturbation bound ϵ, learning rate τ, ascent steps K, ascent step size α

Initialize $\boldsymbol{\theta}$
for epoch $= 1 \ldots N_{ep}$ **do**
  for minibatch $B \subset X$ **do**
    $\boldsymbol{\delta}_0 \leftarrow \frac{1}{\sqrt{N_{\boldsymbol{\delta}}}} U(-\epsilon, \epsilon)$, $\boldsymbol{g}_0 \leftarrow 0$
    for $t = 1 \ldots K$ **do**
      Accumulate gradient of parameters $\boldsymbol{\theta}$ given $\boldsymbol{\delta}_{img,t-1}$ and $\boldsymbol{\delta}_{txt,t-1}$
        $\boldsymbol{g}_t \leftarrow \boldsymbol{g}_{t-1} + \frac{1}{K} \mathbb{E}_{(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt}, \boldsymbol{y}) \in B} [\nabla_{\boldsymbol{\theta}} (\mathcal{L}_{std}(\boldsymbol{\theta}) + \mathcal{R}_{at}(\boldsymbol{\theta}) + \mathcal{R}_{kl}(\boldsymbol{\theta}))]$
      Update the perturbation $\boldsymbol{\delta}_{img}$ and $\boldsymbol{\delta}_{txt}$ via gradient ascend
        $\tilde{\boldsymbol{y}} = f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt})$
        $\boldsymbol{g}_{img} \leftarrow \nabla_{\boldsymbol{\delta}_{img}} [L(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img} + \boldsymbol{\delta}_{img}, \boldsymbol{x}_{txt}), \boldsymbol{y}) + L_{kl}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img} + \boldsymbol{\delta}_{img}, \boldsymbol{x}_{txt}), \tilde{\boldsymbol{y}})]$
        $\boldsymbol{\delta}_{img,t} \leftarrow \Pi_{\|\boldsymbol{\delta}_{img}\|_F \leq \epsilon} (\boldsymbol{\delta}_{img,t-1} + \alpha \cdot \boldsymbol{g}_{img} / \|\boldsymbol{g}_{img}\|_F)$
        $\boldsymbol{g}_{txt} \leftarrow \nabla_{\boldsymbol{\delta}_{txt}} [L(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt} + \boldsymbol{\delta}_{txt}), \boldsymbol{y}) + L_{kl}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{img}, \boldsymbol{x}_{txt} + \boldsymbol{\delta}_{txt}), \tilde{\boldsymbol{y}})]$
        $\boldsymbol{\delta}_{txt,t} \leftarrow \Pi_{\|\boldsymbol{\delta}_{txt}\|_F \leq \epsilon} (\boldsymbol{\delta}_{txt,t-1} + \alpha \cdot \boldsymbol{g}_{txt} / \|\boldsymbol{g}_{txt}\|_F)$
    end for
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \tau \boldsymbol{g}_K$
  end for
end for

Accumulate the parameter gradient for "free"

Perturbation update via PGD (Projected Gradient Descent)

Parameter update via SGD (Stochastic Gradient Descent)

Results (VQA, VCR, NLVR2, SNLI-VE)

- Established new state of the art on all the tasks considered
- Gain: +0.85 on VQA, +2.9 on VCR, +1.49 on NLVR2, +0.64 on SNLI-VE

Method	VQA		VCR			NLVR^2		SNLI-VE	
	test-dev	test-std	Q→A	QA→R	Q→AR	dev	test-P	val	test
ViLBERT	70.55	70.92	72.42 (73.3)	74.47 (74.6)	54.04 (54.8)	-	-	-	-
VisualBERT	70.80	71.00	70.8 (71.6)	73.2 (73.2)	52.2 (52.4)	67.4	67.0	-	-
LXMERT	72.42	72.54	-	-	-	74.90	74.50	-	-
Unicoder-VL	-	-	72.6 (73.4)	74.5 (74.4)	54.4 (54.9)	-	-	-	-
12-in-1	73.15	-	-	-	-	-	78.87	-	76.95
VL-BERT$_{BASE}$	71.16	-	73.8 (-)	74.4 (-)	55.2 (-)	-	-	-	-
Oscar$_{BASE}$	73.16	73.44	-	-	-	78.07	78.36	-	-
UNITER$_{BASE}$	72.70	72.91	74.56 (75.0)	77.03 (77.2)	57.76 (58.2)	77.18	77.85	78.59	78.28
VILLA$_{BASE}$	**73.59**	**73.67**	**75.54 (76.4)**	**78.78 (79.1)**	**59.75 (60.6)**	**78.39**	**79.30**	**79.47**	**79.03**
VL-BERT$_{LARGE}$	71.79	72.22	75.5 (75.8)	77.9 (78.4)	58.9 (59.7)	-	-	-	-
Oscar$_{LARGE}$	73.61	73.82	-	-	-	79.12	80.37	-	-
UNITER$_{LARGE}$	73.82	74.02	77.22 (77.3)	80.49 (80.8)	62.59 (62.8)	79.12	79.98	79.39	79.38
VILLA$_{LARGE}$	**74.69**	**74.87**	**78.45 (78.9)**	**82.57 (82.8)**	**65.18 (65.7)**	**79.76**	**81.47**	**80.18**	**80.02**

(a) Results on VQA, VCR, NLVR^2, and SNLI-VE.

Results (ITR, RE)

- Gain: +1.52/+0.60 on Flickr30k IR & TR (R@1), and +0.99 on RE

Method	RefCOCO+						RefCOCO					
	val	testA	testB	val^d	testA^d	testB^d	val	testA	testB	val^d	testA^d	testB^d
ViLBERT	-	-	-	72.34	78.52	62.61	-	-	-	-	-	-
VL-BERT$_{BASE}$	79.88	82.40	75.01	71.60	77.72	60.99	-	-	-	-	-	-
UNITER$_{BASE}$	83.66	86.19	78.89	75.31	81.30	65.58	91.64	92.26	90.46	81.24	86.48	73.94
VILLA$_{BASE}$	**84.26**	**86.95**	**79.22**	**76.05**	**81.65**	**65.70**	**91.93**	**92.79**	**91.38**	**81.65**	**87.40**	**74.48**
VL-BERT$_{LARGE}$	80.31	83.62	75.45	72.59	78.57	62.30	-	-	-	-	-	-
UNITER$_{LARGE}$	84.25	**86.34**	79.75	75.90	81.45	66.70	91.84	92.65	91.19	81.41	87.04	74.17
VILLA$_{LARGE}$	**84.40**	86.22	**80.00**	**76.17**	**81.54**	**66.84**	**92.58**	**92.96**	**91.62**	**82.39**	**87.48**	**74.84**

(b) Results on RefCOCO+ and RefCOCO. The superscript d denotes evaluation using detected proposals.

Method	RefCOCOg				Flickr30k IR			Flickr30k TR		
	val	test	val^d	test^d	R@1	R@5	R@10	R@1	R@5	R@10
ViLBERT	-	-	-	-	58.20	84.90	91.52	-	-	-
Unicoder-VL	-	-	-	-	71.50	90.90	94.90	86.20	96.30	99.00
UNITER$_{BASE}$	86.52	86.52	74.31	74.51	72.52	92.36	**96.08**	85.90	97.10	98.80
VILLA$_{BASE}$	**88.13**	**88.03**	**75.90**	**75.93**	**74.74**	**92.86**	95.82	**86.60**	**97.90**	**99.20**
UNITER$_{LARGE}$	87.85	87.73	74.86	75.77	75.56	94.08	96.76	87.30	**98.00**	**99.20**
VILLA$_{LARGE}$	**88.42**	**88.97**	**76.18**	**76.71**	**76.26**	**94.24**	**96.84**	**87.90**	97.50	98.80

(c) Results on RefCOCOg and Flickr30k Image Retrieval (IR) and Text Retrieval (TR).

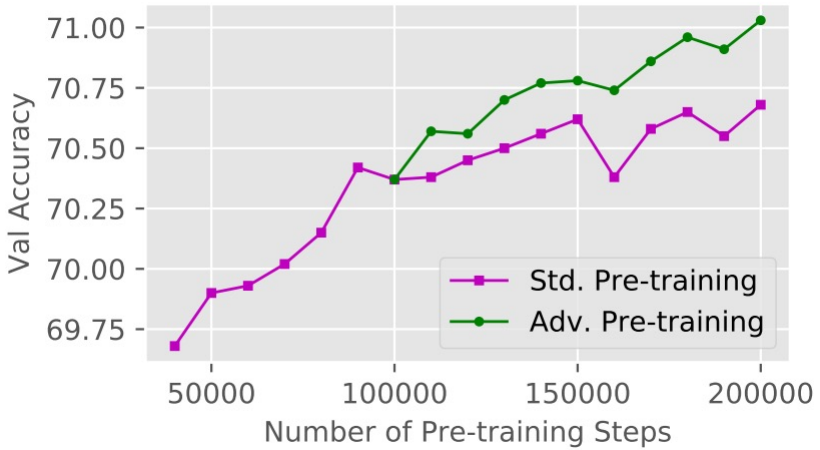
Ablation study

- Both adversarial pre-training and finetuning contribute to performance boost

Method	VQA	VCR (val)			$NLVR^2$	VE	Flickr30k IR			RefCOCO		Ave.	
	test-dev	Q→A	QA→R	Q→AR	test-P	test	R@1	R@5	R@10	$testA^d$	$testB^d$		
UNITER (reimp.)	72.70	74.24	76.93	57.31	77.85	78.28	72.52	92.36	96.08	86.48	73.94	78.06	*+0.51*
VILLA-pre	73.03	74.76	77.04	57.82	78.44	78.43	73.76	93.02	96.28	87.34	74.35	78.57	*+0.82*
VILLA-fine	73.29	75.18	78.29	59.08	78.84	78.86	73.46	92.98	96.26	87.17	74.31	78.88	*+1.15*
VILLA	73.59	75.54	78.78	59.75	79.30	79.03	74.74	92.86	95.82	87.40	74.48	**79.21**	

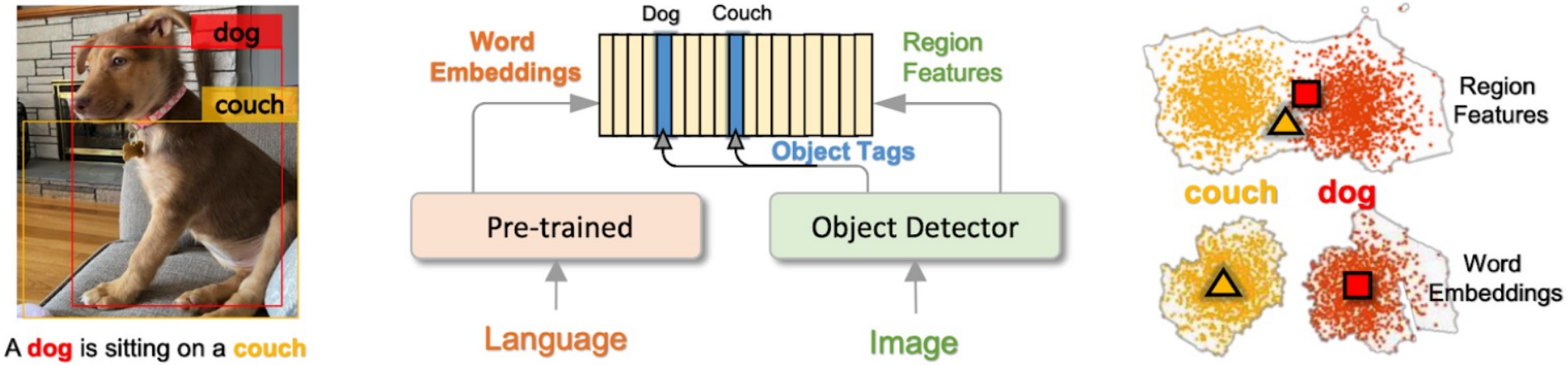
- VILLA can be applied to any VLP models

Method	VQA		GQA		$NLVR^2$		Meta-Ave.
	test-dev	test-std	test-dev	test-std	dev	test-P	
LXMERT	72.42	72.54	60.00	60.33	74.95	74.45	69.12
LXMERT (reimp.)	72.50	72.52	59.92	60.28	74.72	74.75	69.12
VILLA-fine	**73.02**	**73.18**	**60.98**	**61.12**	**75.98**	**75.73**	**70.00**



OSCAR: Object Tags as Anchor Points

- How OSCAR represents an image-text pair into semantic space

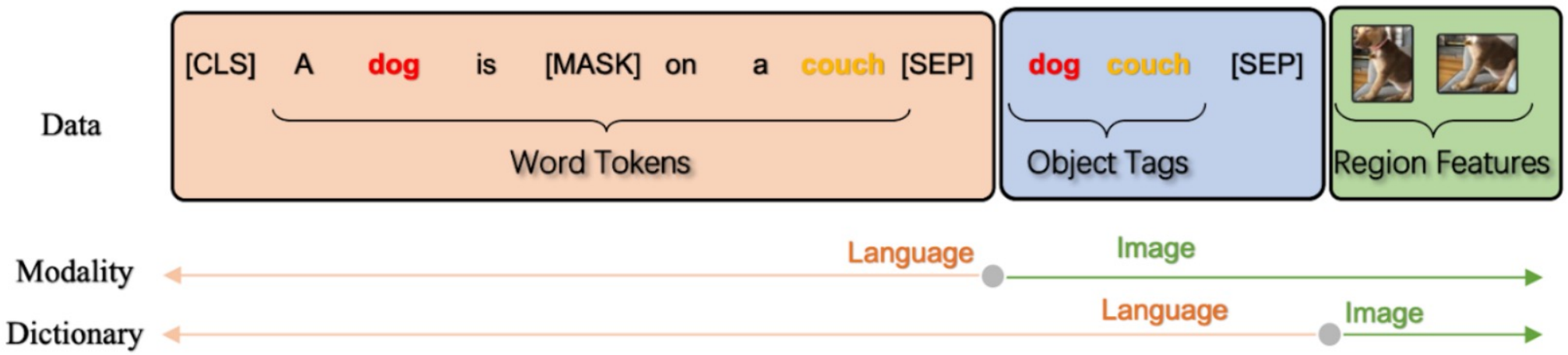


(a) Image-text pair (b) Objects as anchor points (c) Semantics spaces

[1 Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks, ECCV 2020

OSCAR: Object Tags as Anchor Points

- The image-text pair is represented as a triplet



[1 Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks, ECCV 2020

OSCAR: Object Tags as Anchor Points

- OSCAR improves the cross-domain alignment

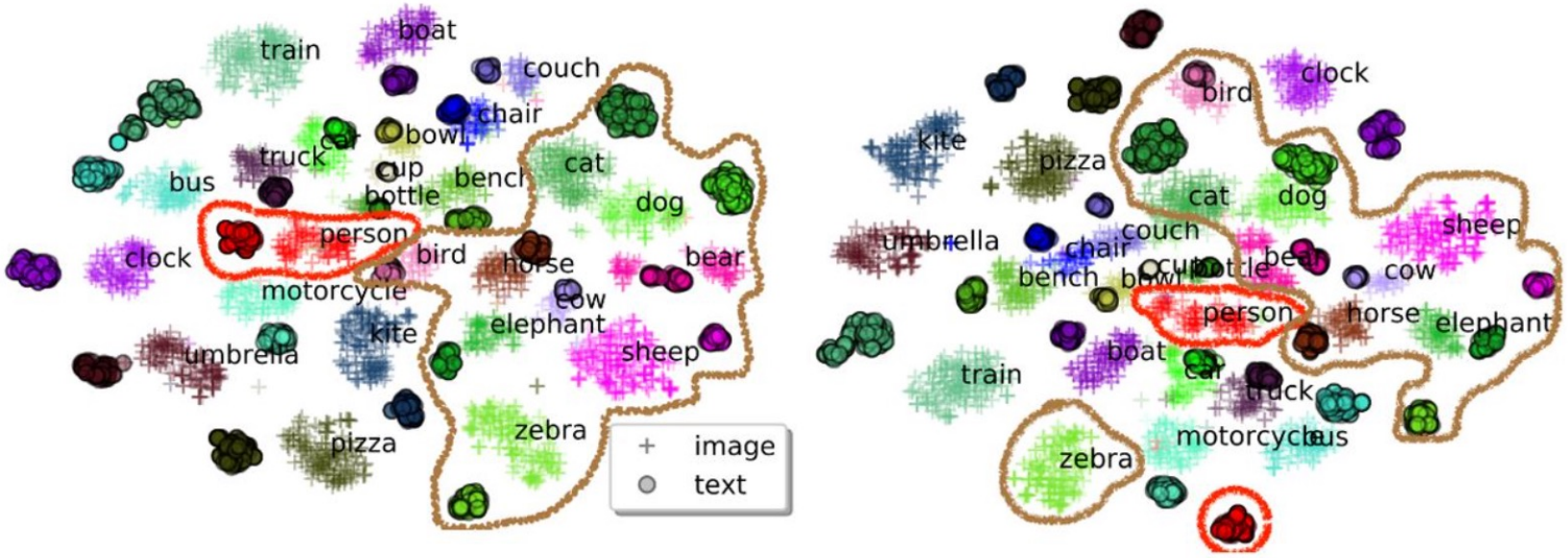
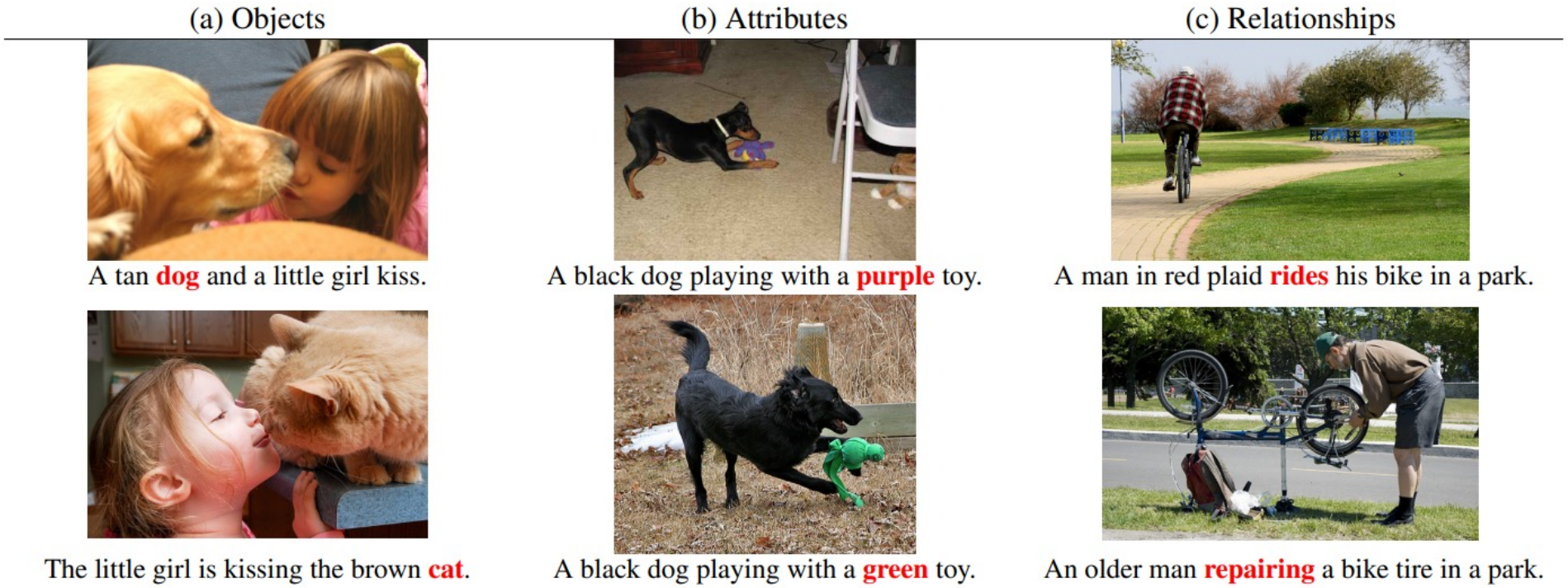


Figure 4: 2D visualization using t-SNE. The points from the same object class share the same color. Oscar (left) improves the cross-domain alignment over the baseline without object tags (right). Red and grey curves cover the objects of the same and related semantics, respectively.

[1 Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks, ECCV 2020

ERINE-ViL: Scene Graph Prediction

- How to learn fine-grained semantics during pre-training



[1 ERNIE-ViL: Knowledge Enhanced Vision-Language Representations Through Scene Graph, AAAI 2021

ERINE-ViL: Scene Graph Prediction

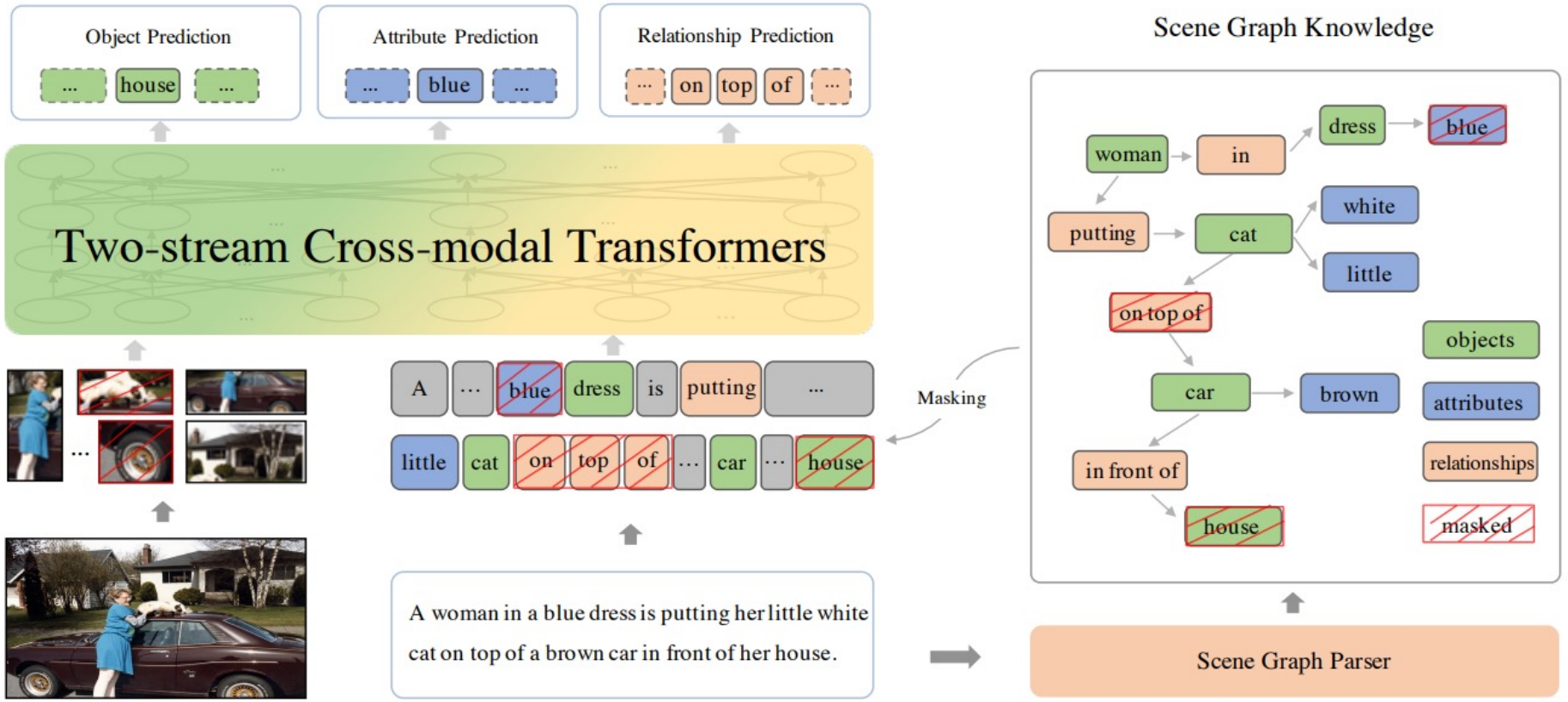
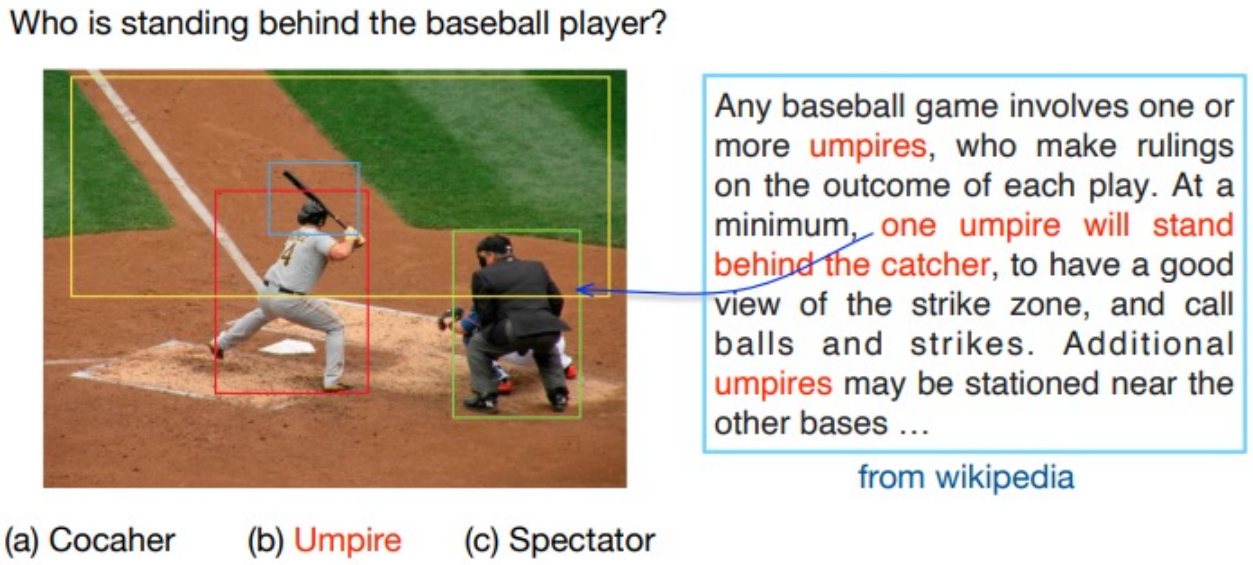


Figure 2: Illustration of Scene Graph Prediction tasks for ERNIE-ViL. Given detected regions of the image and token sequence of the text, ERNIE-ViL uses a two-stream cross-modal Transformers network to model the joint vision-language representations. Based on the scene graph parsed from the text using Scene Graph Parser, we construct Object Prediction, Attribute Prediction and Relationship Prediction tasks to learn cross-modal detailed semantics alignments.

[1 ERNIE-ViL: Knowledge Enhanced Vision-Language Representations Through Scene Graph, AAAI 2021

UNIMO: Unified-modal learning

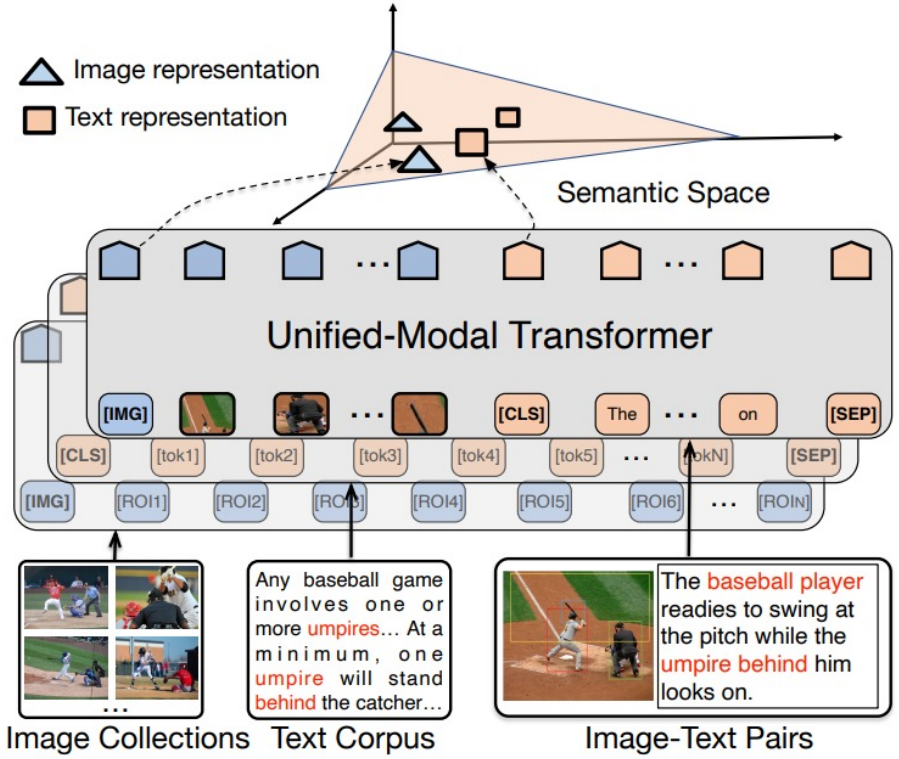
- We can only determine the correct answer to the visual question based on the textual background information



[1 UNIMO: Towards Unified-Modal Understanding and Generation via Cross-Modal Contrastive Learning, ACL 2021

UNIMO: Unified-modal learning

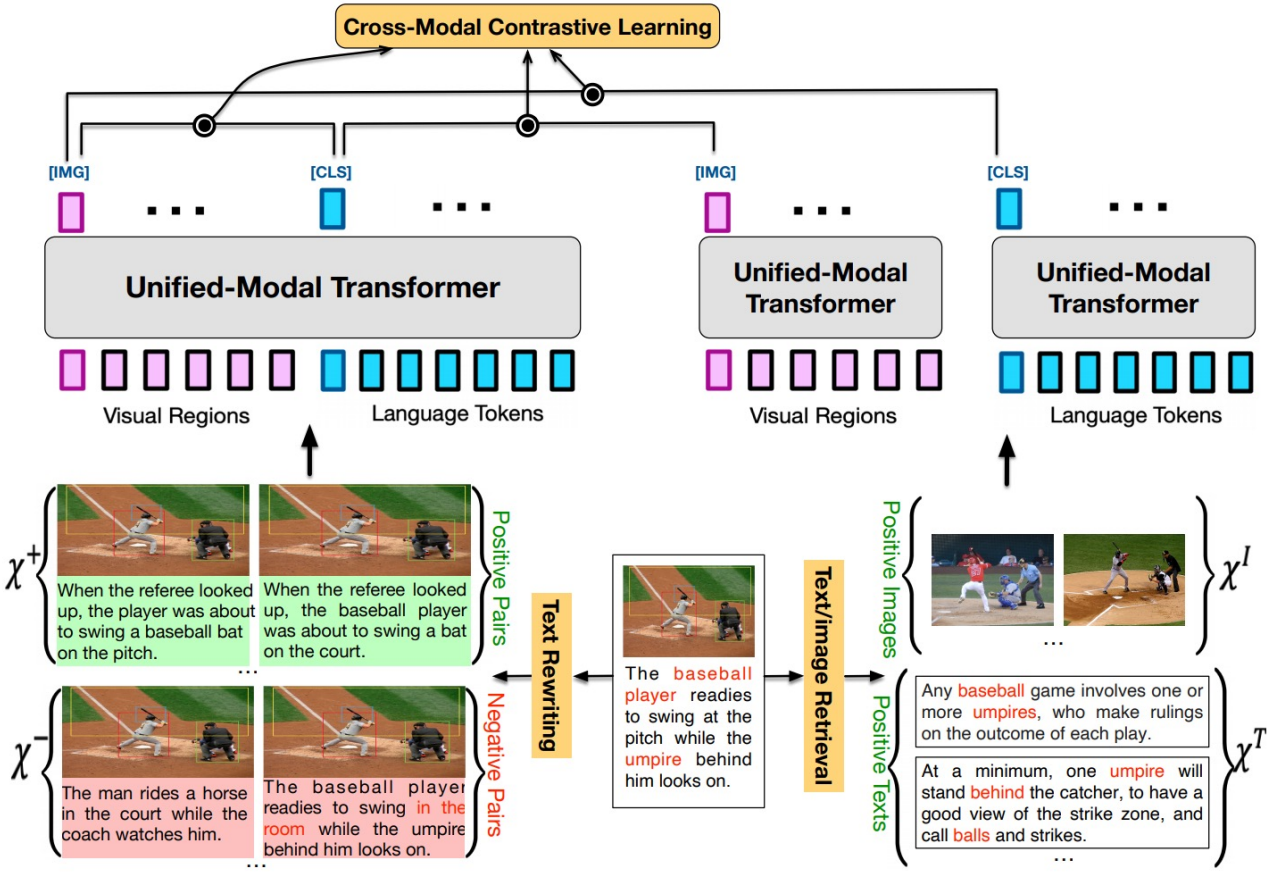
- Both image collections, text corpus, and image-text pairs can be effectively utilized for pre-training



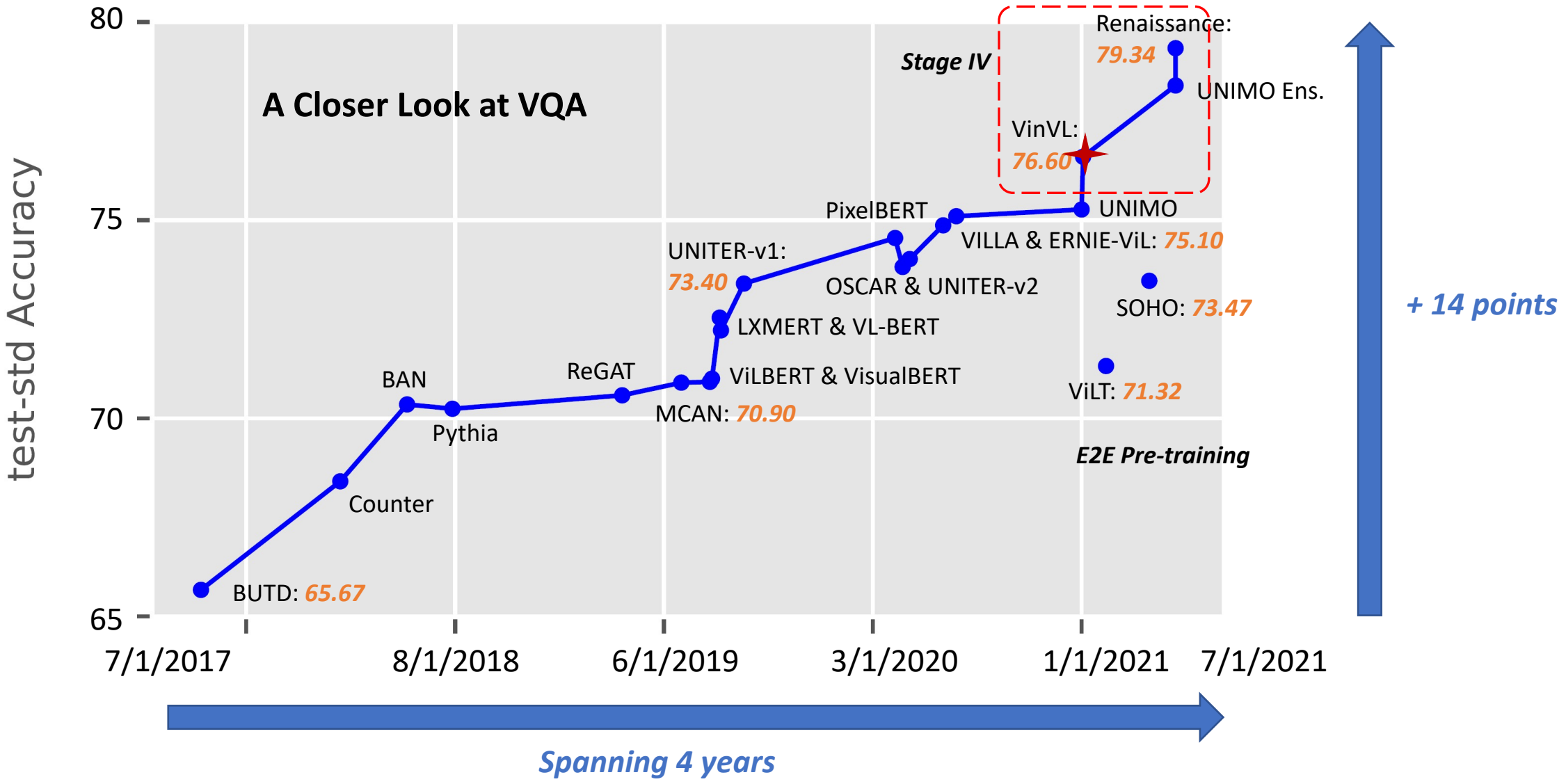
[1 UNIMO: Towards Unified-Modal Understanding and Generation via Cross-Modal Contrastive Learning, ACL 2021

UNIMO: Unified-modal learning

- Using cross-modal contrastive learning as a pre-training task

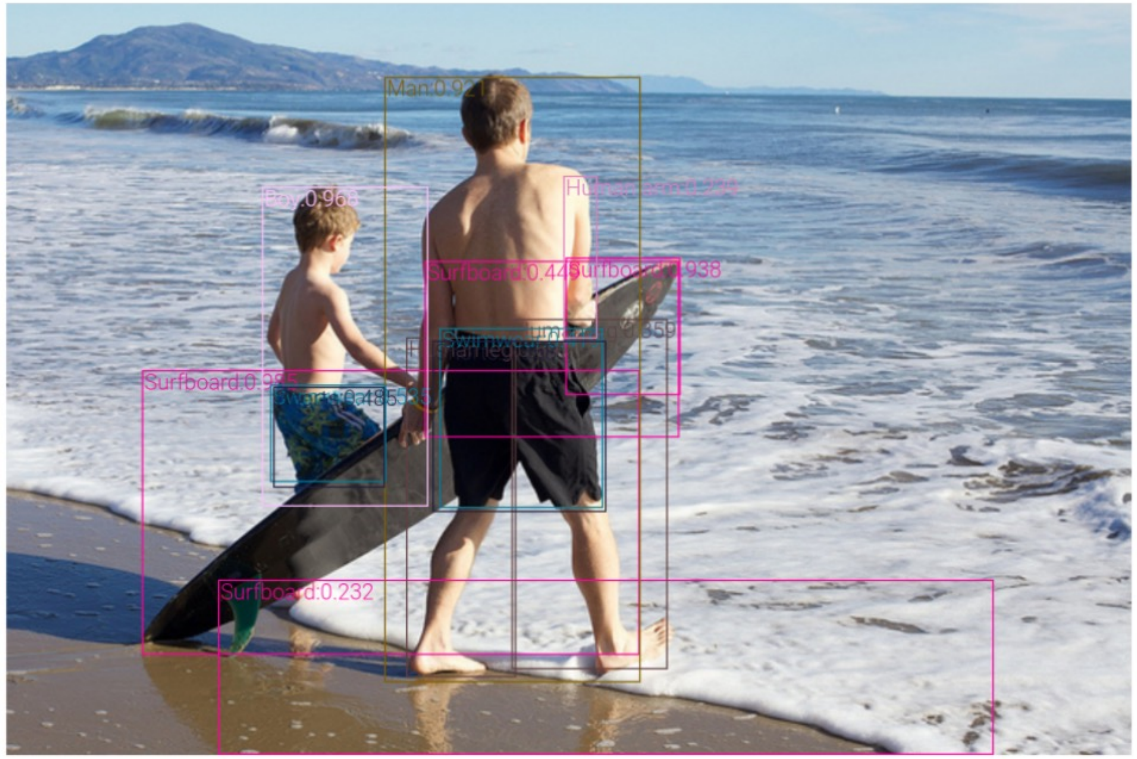


[1 UNIMO: Towards Unified-Modal Understanding and Generation via Cross-Modal Contrastive Learning, ACL 2021


A Closer Look at VQA
test-std Accuracy
80
75
70
65
7/1/2017
8/1/2018
6/1/2019
3/1/2020
1/1/2021
7/1/2021
BUTD: 65.67
Counter
BAN
Pythia
ReGAT
MCAN: 70.90
ViLBERT & VisualBERT
LXMERT & VL-BERT
UNITER-v1: 73.40
OSCAR & UNITER-v2
PixelBERT
VILLA & ERNIE-ViL: 75.10
UNIMO
VinVL: 76.60
Stage IV
Renaissance: 79.34
UNIMO Ens.
SOHO: 73.47
ViLT: 71.32
E2E Pre-training
+ 14 points
Spanning 4 years

VinVL: Revisiting Visual Features in VL Models

- VinVL captures much richer image semantics



X152-FPN model trained on OpenImages

X152-C4 model trained on four public object detection datasets

[1 VinVL: Revisiting Visual Representations in Vision-Language Models, CVPR 2021

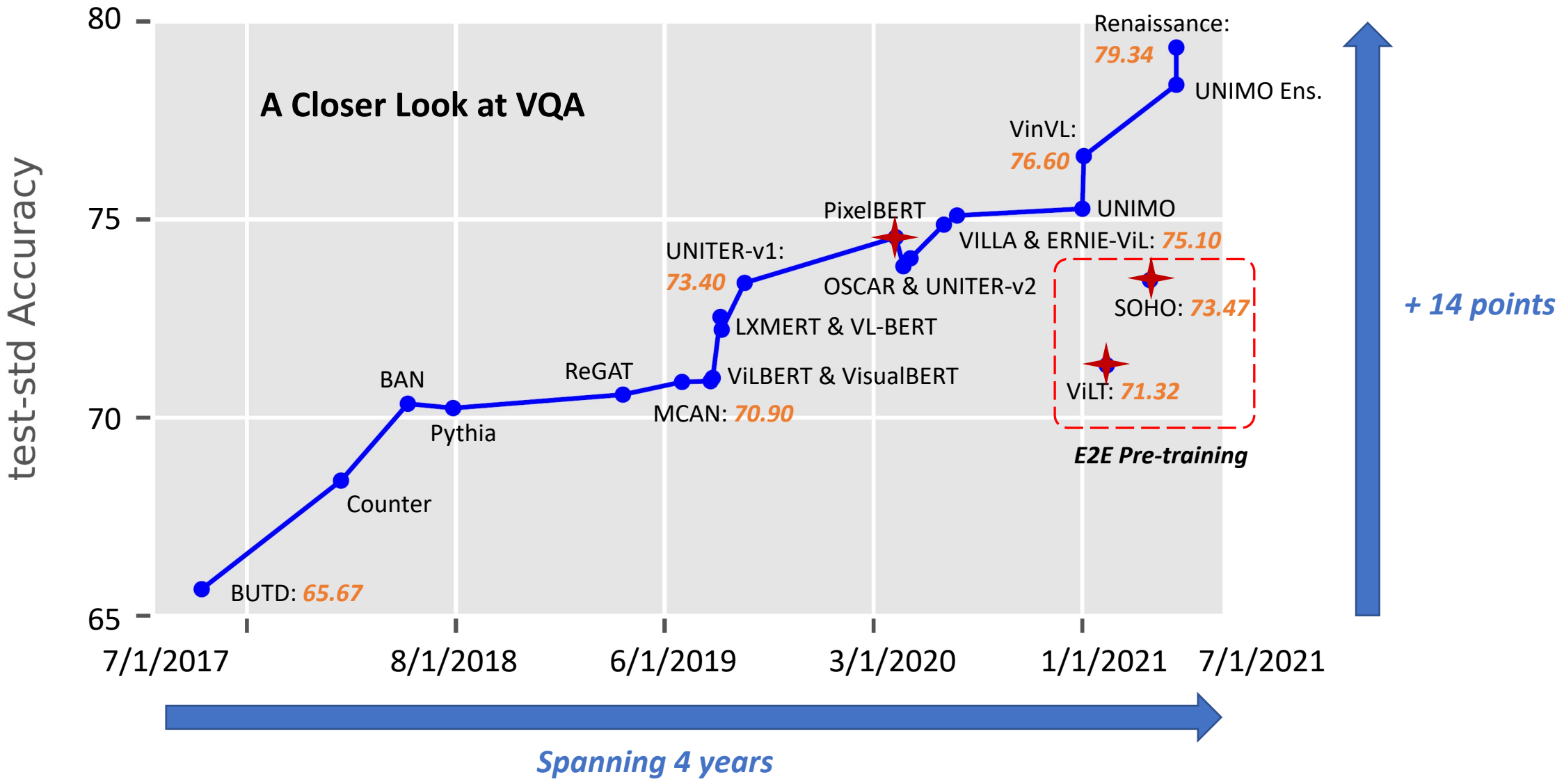
VinVL: Revisiting Visual Features in VL Models

- We know visual feature matters, yet we never know it matters so significantly!

Visual feature	VQA		GQA		Image Captioning				NoCaps		Image Retrieval			Text Retrieval			NLVR2	
	test-dev	test-std	test-dev	test-std	B@4	M	C	S	C	S	R@1	R@5	R@10	R@1	R@5	R@10	dev	test-P
Anderson *et al.* [2]	73.16	73.44	61.58	61.62	40.5	29.7	137.6	22.8	86.58	12.38	54.0	80.8	88.5	70.0	91.1	95.5	78.07	78.36
Ours	**75.95**	**76.12**	**65.05**	**64.65**	**40.9**	**30.9**	**140.6**	**25.1**	**92.46**	**13.07**	**58.1**	**83.2**	**90.1**	**74.6**	**92.6**	**96.3**	**82.05**	**83.08**
Δ	**2.79** ↑	**2.68** ↑	**3.47** ↑	**3.03** ↑	**0.4** ↑	**1.2** ↑	**3.0** ↑	**2.3** ↑	**5.9** ↑	**0.7** ↑	**4.1** ↑	**2.4** ↑	**1.6** ↑	**4.6** ↑	**1.5** ↑	**0.8** ↑	**3.98** ↑	**4.71** ↑

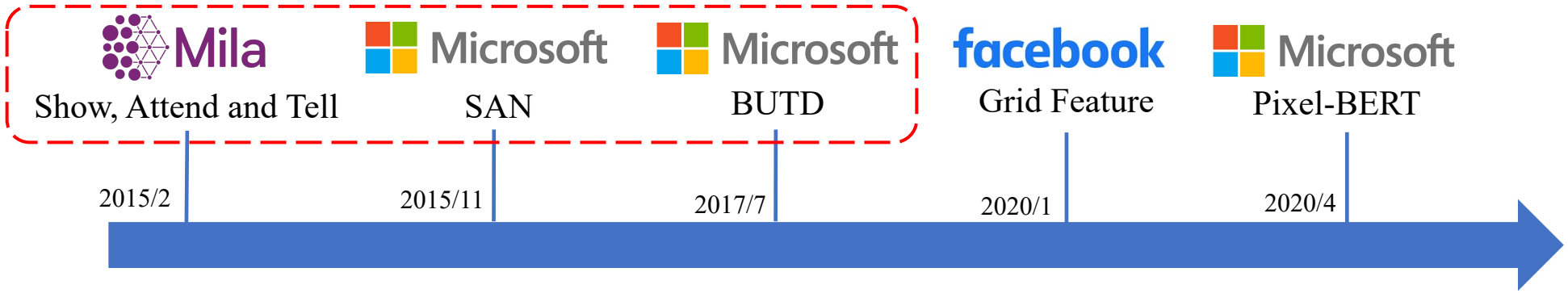
Table 1: Uniform improvements on seven VL tasks by replacing visual features from Anderson *et al.* [2] with ours. The NoCaps baseline is from VIVO [9], and our results are obtained by directly replacing the visual features. The baselines for rest tasks are from OSCAR [21], and our results are obtained by replacing the visual features and performing OSCAR+ pre-training. All models are BERT-Base size. As analyzed in Section 5.2, the new visual features contributes 95% of the improvement.

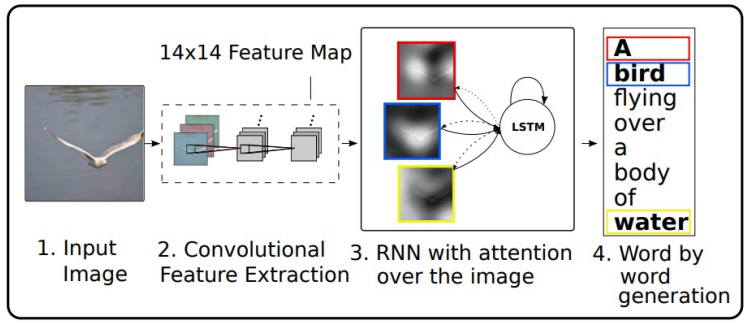
[1 VinVL: Revisiting Visual Representations in Vision-Language Models, CVPR 2021


A Closer Look at VQA
test-std Accuracy
80
75
70
65
7/1/2017
8/1/2018
6/1/2019
3/1/2020
1/1/2021
7/1/2021
BUTD: 65.67
Counter
BAN
Pythia
ReGAT
MCAN: 70.90
ViLBERT & VisualBERT
LXMERT & VL-BERT
UNITER-v1:
73.40
OSCAR & UNITER-v2
PixelBERT
VILLA & ERNIE-ViL: 75.10
UNIMO
VinVL:
76.60
Renaissance:
79.34
UNIMO Ens.
SOHO: 73.47
ViLT: 71.32
E2E Pre-training
+ 14 points
Spanning 4 years

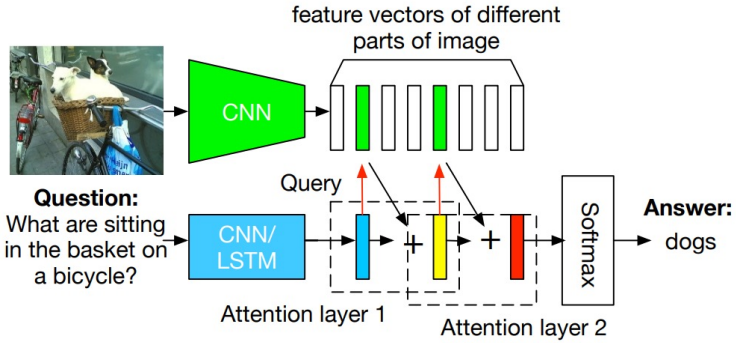
The history of using grid features

- From *grid* features to *region* features, and to *grid* features again

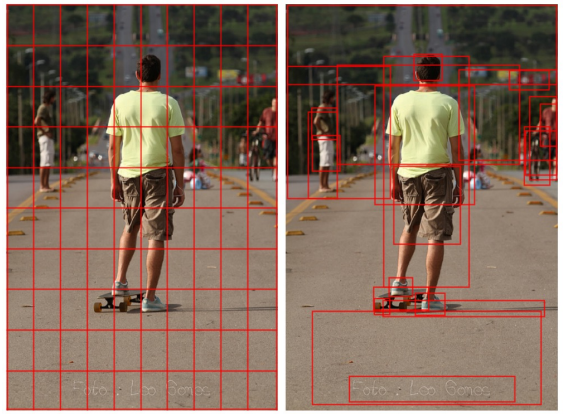




Show, Attend and Tell



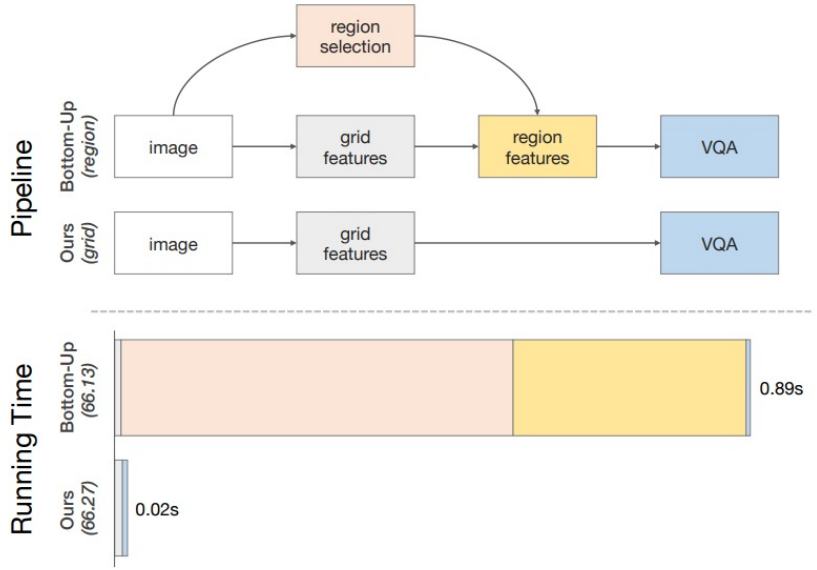
Stacked Attention Network



Bottom-Up Top-Down Attention

In Defense of Grid Features

- *Findings*: Using grid features is fast, and it can achieve comparable performance with regional features
- But why previous methods based on grid features cannot outperform BUTD features? *Two reasons*: Pre-training task, and input image size

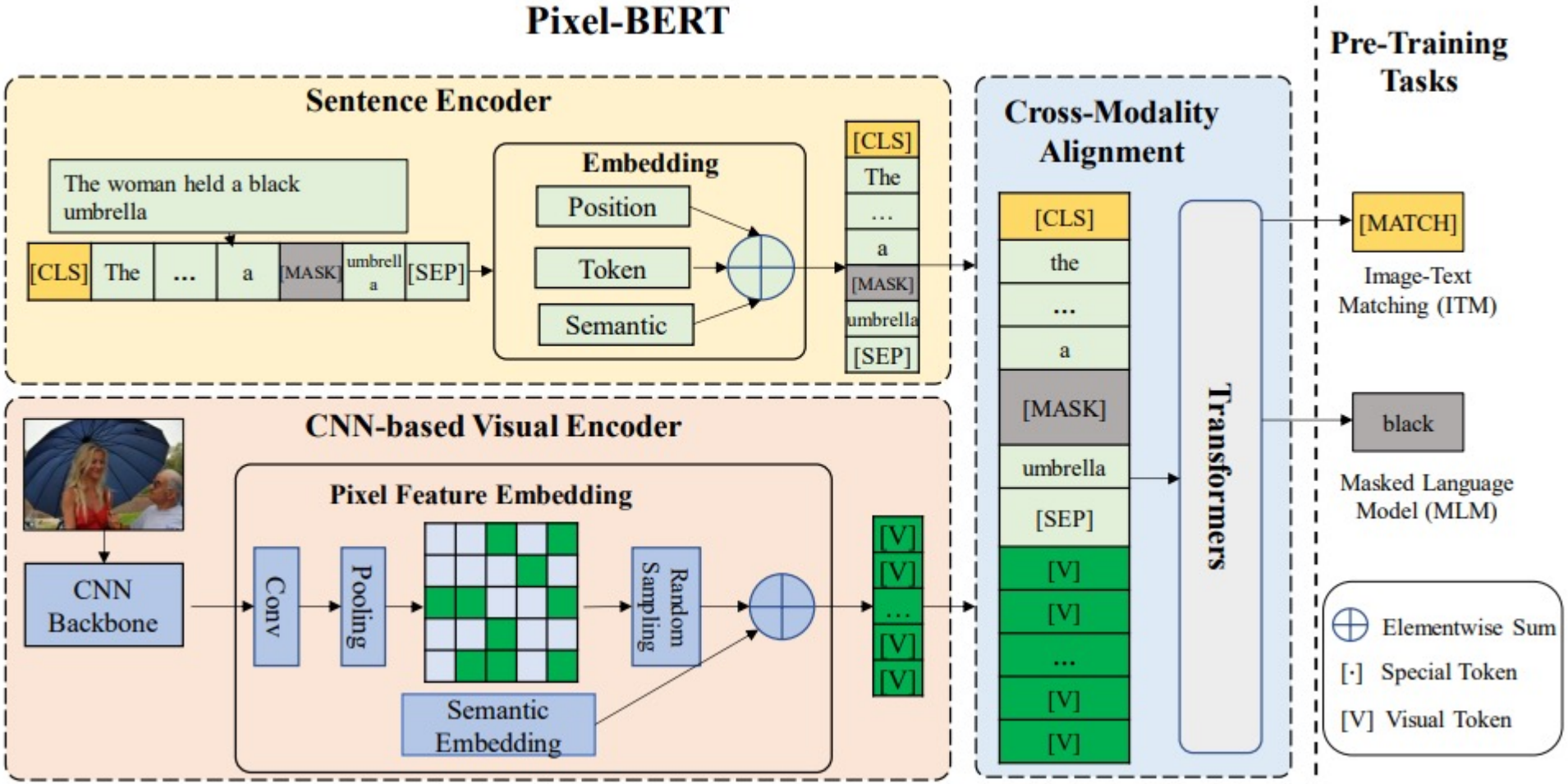


		accuracy	pre-training task	input size
G	*prev.*	60.76	ImageNet [6] classification	448×448
	ours	64.37	VG [22] object+attribute detection	600×1000

Table 3: Comparison between the conventional **ImageNet pre-trained and our proposed grid features** on the VQA 2.0 `vqa-eval` set. Besides VQA accuracy, we list two major differences between the two: 1) pre-training task and 2) input image size.

[1] In Defense of Grid Features for Visual Question Answering, CVPR 2020

Pixel-BERT: An E2E Pre-training Framework



[1] Pixel-BERT: Aligning Image Pixels with Text by Deep Multi-Modal Transformers, 2020

Pixel-BERT: An E2E Pre-training Framework

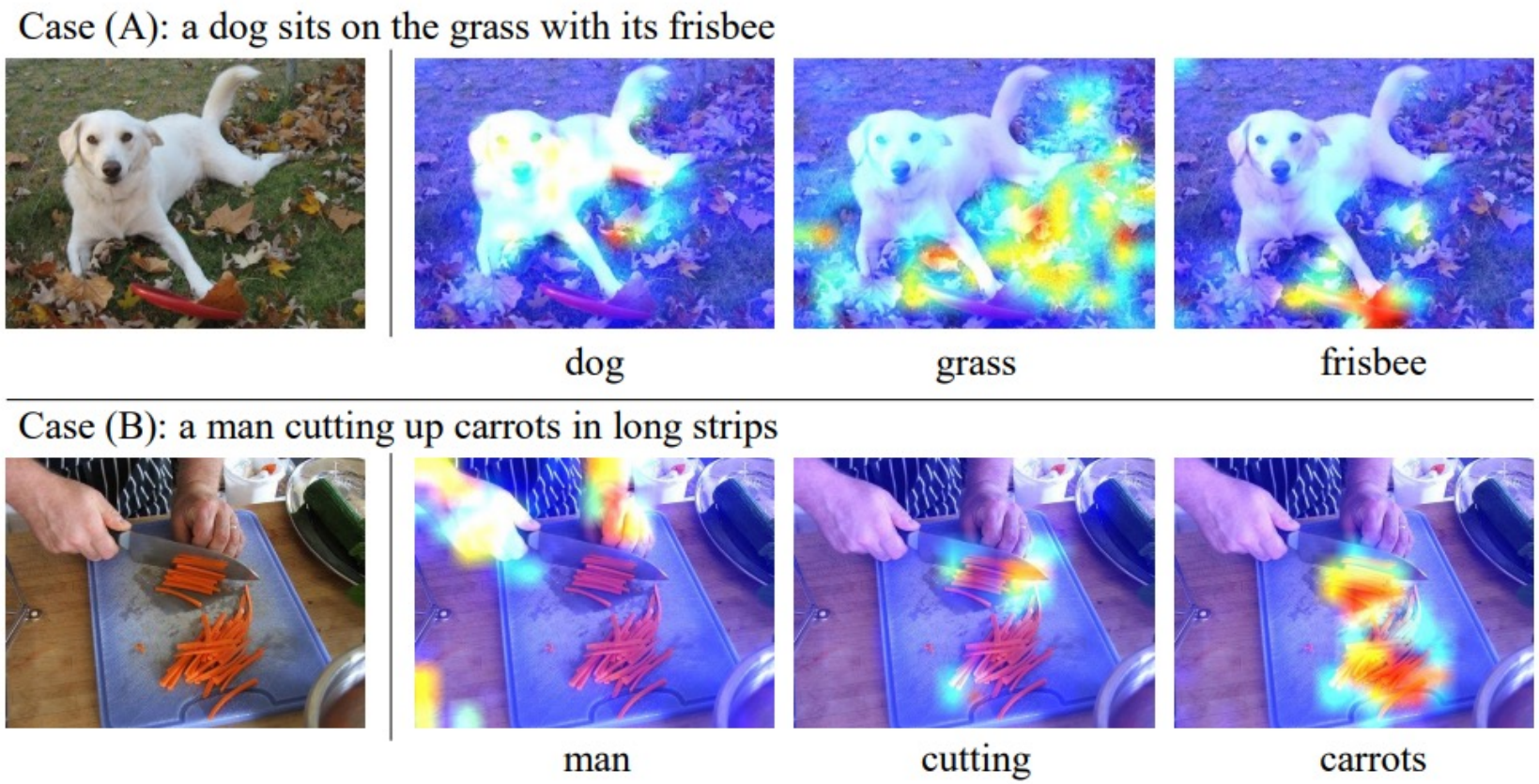
- It can achieve very good performance with a strong vision backbone.

Model	test-dev	test-std
MUTAN[5]	60.17	-
BUTD[2]	65.32	65.67
ViLBERT[21]	70.55	70.92
VisualBERT[19]	70.80	71.00
VLBERT[29]	71.79	72.22
LXMERT[33]	72.42	72.54
UNITER[6]	72.27	72.46
Pixel-BERT (r50)	71.35	71.42
Pixel-BERT (x152)	**74.45**	**74.55**

Table 2. Evaluation of Pixel-BERT with other methods on VQA.

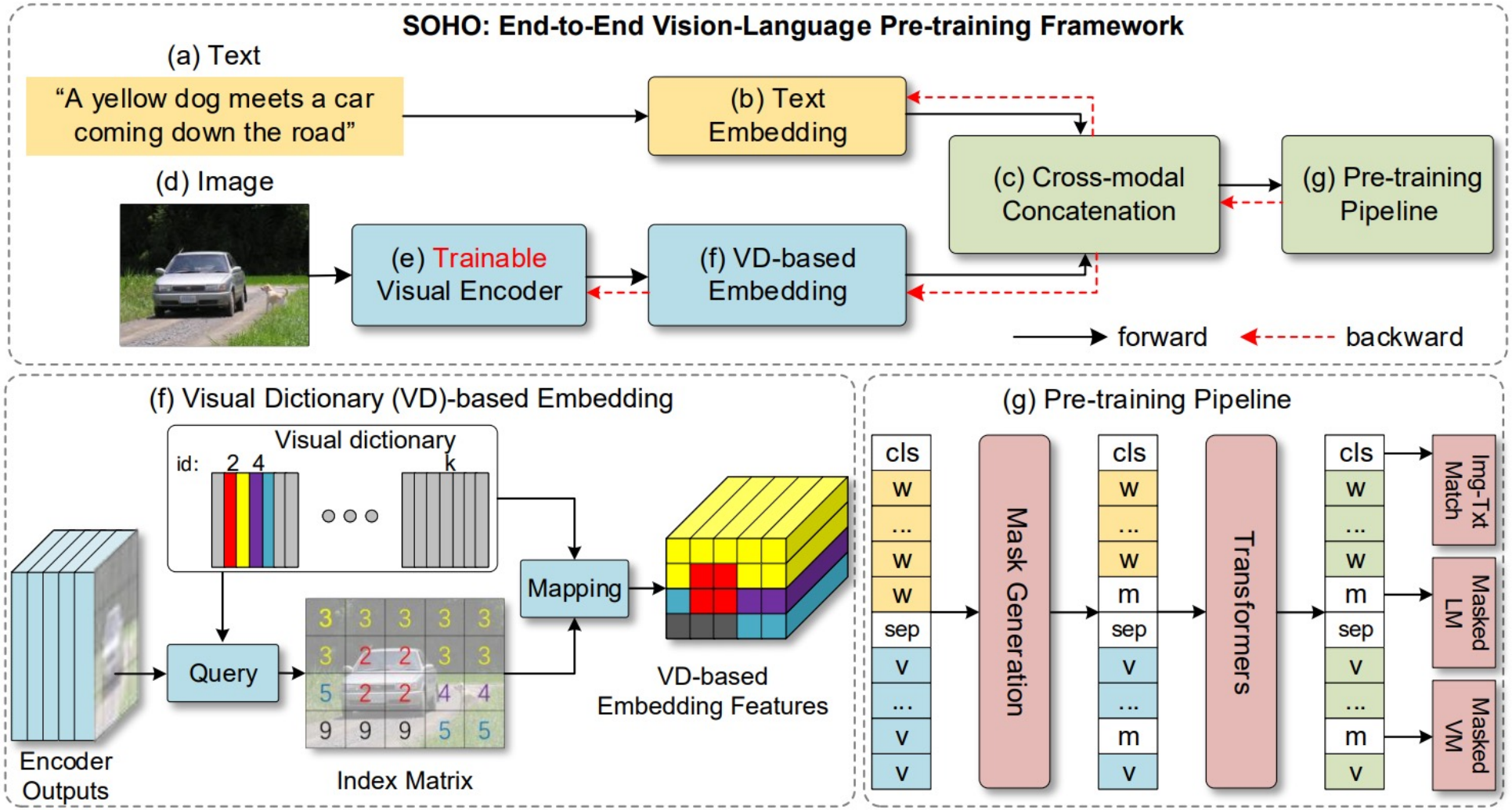
[1] Pixel-BERT: Aligning Image Pixels with Text by Deep Multi-Modal Transformers, 2020

Pixel-BERT Learns Good Word-Pixel Alignment



[1] Pixel-BERT: Aligning Image Pixels with Text by Deep Multi-Modal Transformers, 2020

SOHO: Seeing Out of tHe bOx



[1] Seeing Out of tHe bOx: End-to-End Pre-training for Vision-Language Representation Learning, CVPR 2021

SOHO Results

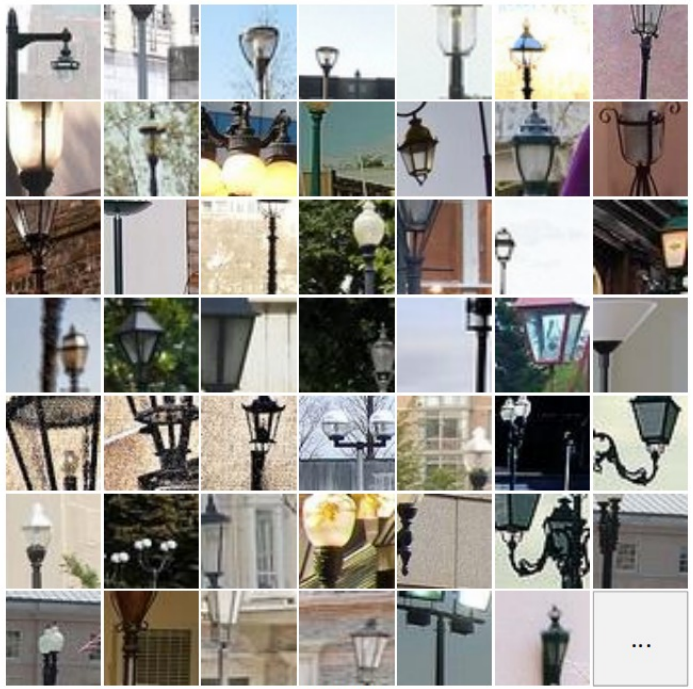
- It can achieve better performance than models using BUTD features
- Visual dictionary is necessary to improve the performance. *But why?*
  - Aggregating similar visual semantics into the same image feature
  - Mimicking the behavior of object detection models
- *10x faster*: SOHO (44ms) vs. BUTD (464ms)

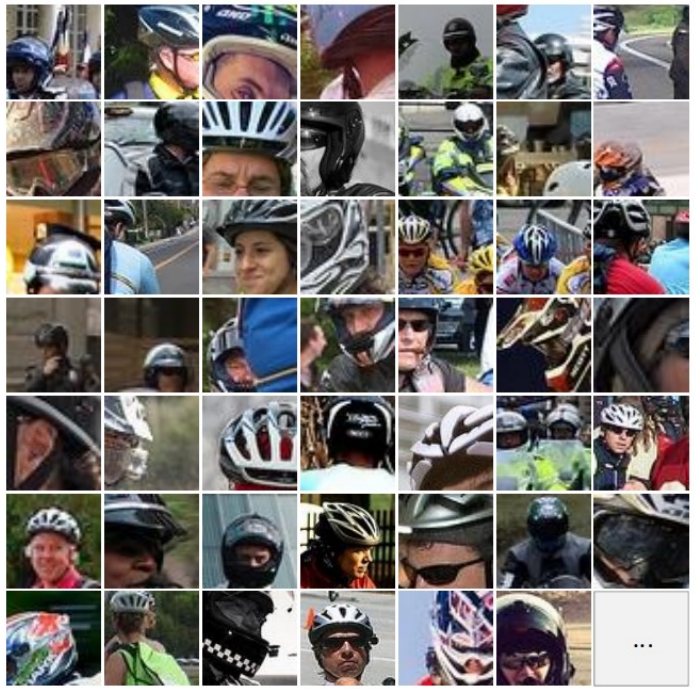
Table 4: Evaluation of VQA on VQA 2.0 dataset. "-" indicates the detail is not reported. X101 denotes ResNeXt-101 architecture [45].

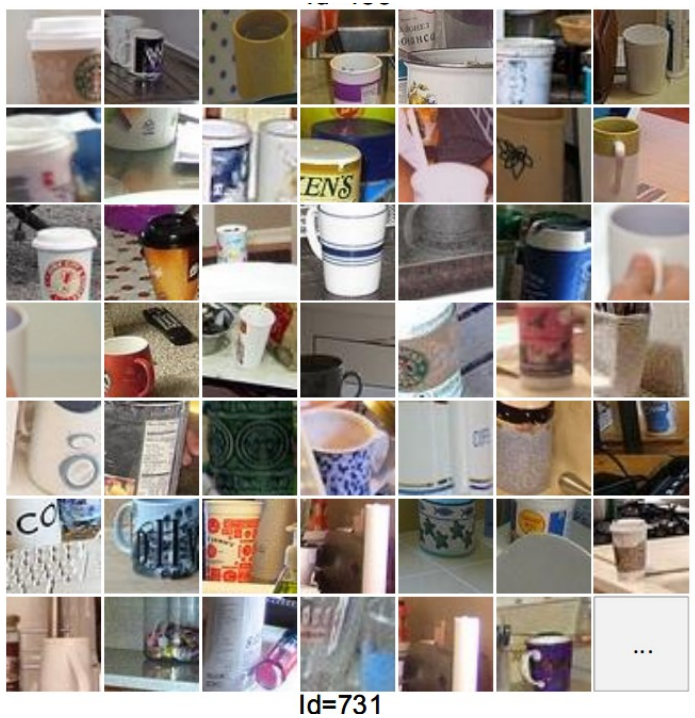
Model	Backbone	test-dev	test-std
MUTAN[4]	R152	60.17	-
BUTD[2]	R101	65.32	65.67
Unified VLP [50]	X101	70.50	70.70
ViLBERT[27]	R101	70.55	70.92
VisualBERT[23]	R152	70.80	71.00
VLBERT[36]	R101	71.79	72.22
LXMERT[39]	R101	72.42	72.54
UNITER[7]	R101	72.70	72.91
SOHO (Ours)	R101	**73.25**	**73.47**

[1] Seeing Out of tHe bOx: End-to-End Pre-training for Vision-Language Representation Learning, CVPR 2021

Visual Dictionary Visualization



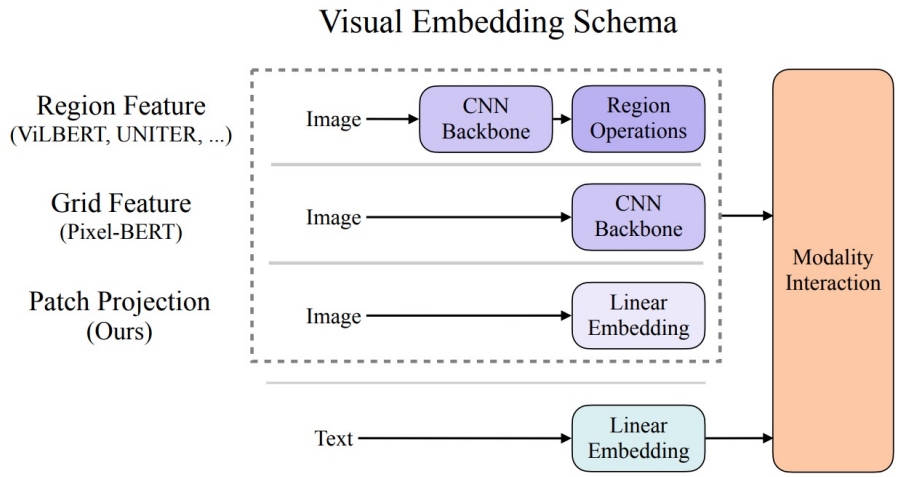


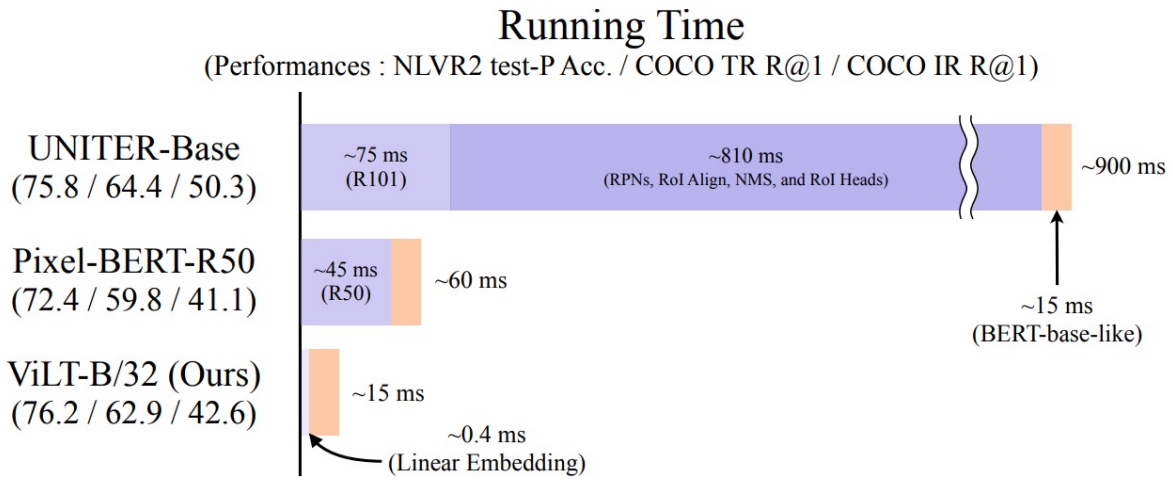


[1] Seeing Out of tHe bOx: End-to-End Pre-training for Vision-Language Representation Learning, CVPR 2021

ViLT: Without Convolution or Region Supervision

- ViLT is very fast since both object detection models and CNNs are not used

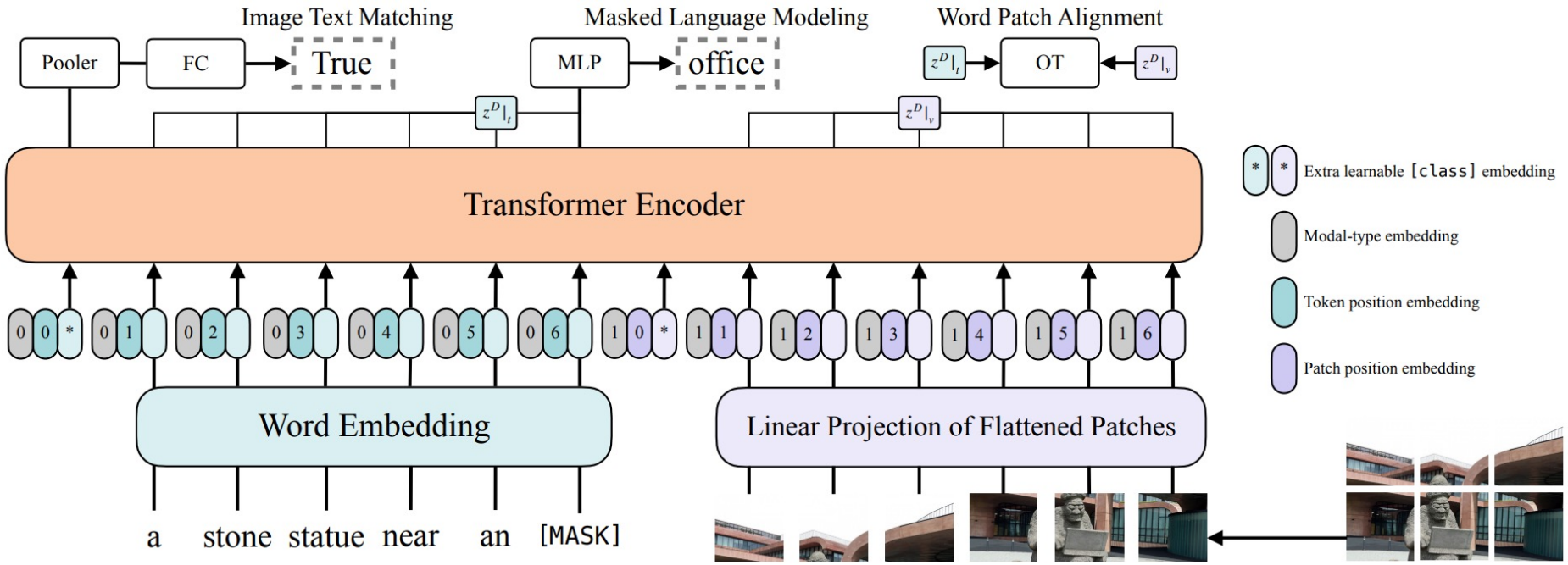




[1] ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision, ICML 2021

ViLT: Without Convolution or Region Supervision

- A single unified transformer is learned



[1] ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision, ICML 2021

ViLT: Without Convolution or Region Supervision

- However, performance-wise, it is still not ideal

Table 2. Comparison of ViLT-B/32 with other models on downstream classification tasks. We use MCAN (Yu et al., 2019) and MaxEnt (Suhr et al., 2018) for VQAv2 and NLVR2 w/o VLP SOTA results. † additionally used GQA, VQAv2, VG-QA for pretraining. ‡ used open images dataset (Kuznetsova et al., 2020) to expand its visual vocabulary of region feature embeddings. ⓐ indicates RandAugment is applied during finetuning.

Visual Embed	Model	Time (ms)	VQAv2 test-dev	NLVR2 dev	NLVR2 test-P
Region	w/o VLP SOTA	~900	70.63	54.80	53.50
	ViLBERT	~920	70.55	-	-
	VisualBERT-Base	~1000	70.80	67.40	67.00
	LXMERT	~910	72.42	74.90	74.50
	UNITER-Base	~900	72.70	75.85	75.80
	OSCAR-Base†	~900	73.16	78.07	78.36
	VinVL-Base†‡	~1000	75.95	82.05	83.08
Grid	Pixel-BERT-X152	~120	74.45	76.50	77.20
	Pixel-BERT-R50	~60	71.35	71.70	72.40
Linear	ViLT-B/32	~15	70.34	74.56	74.66
	ViLT-B/32ⓐ	~15	70.94	75.24	76.21

5 Points Gap
On VQA

- For pre-training, the longer the better
- Whole word masking and *image augmentation* are both useful
- Masked patch prediction (MPP) is not useful

[1] ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision, ICML 2021

ViLT Learns Good Word-Patch Alignment

a display of flowers growing out and over the retaining wall in front of cottages on a cloudy day.


flowers

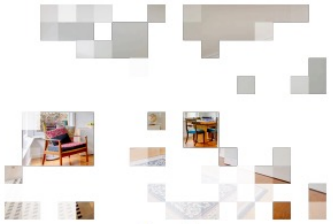

wall


cottages

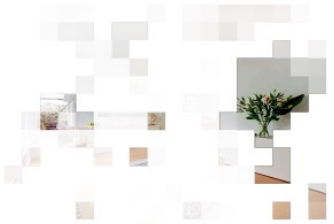

cloudy

a room with a rug, a chair, a painting, and a plant.

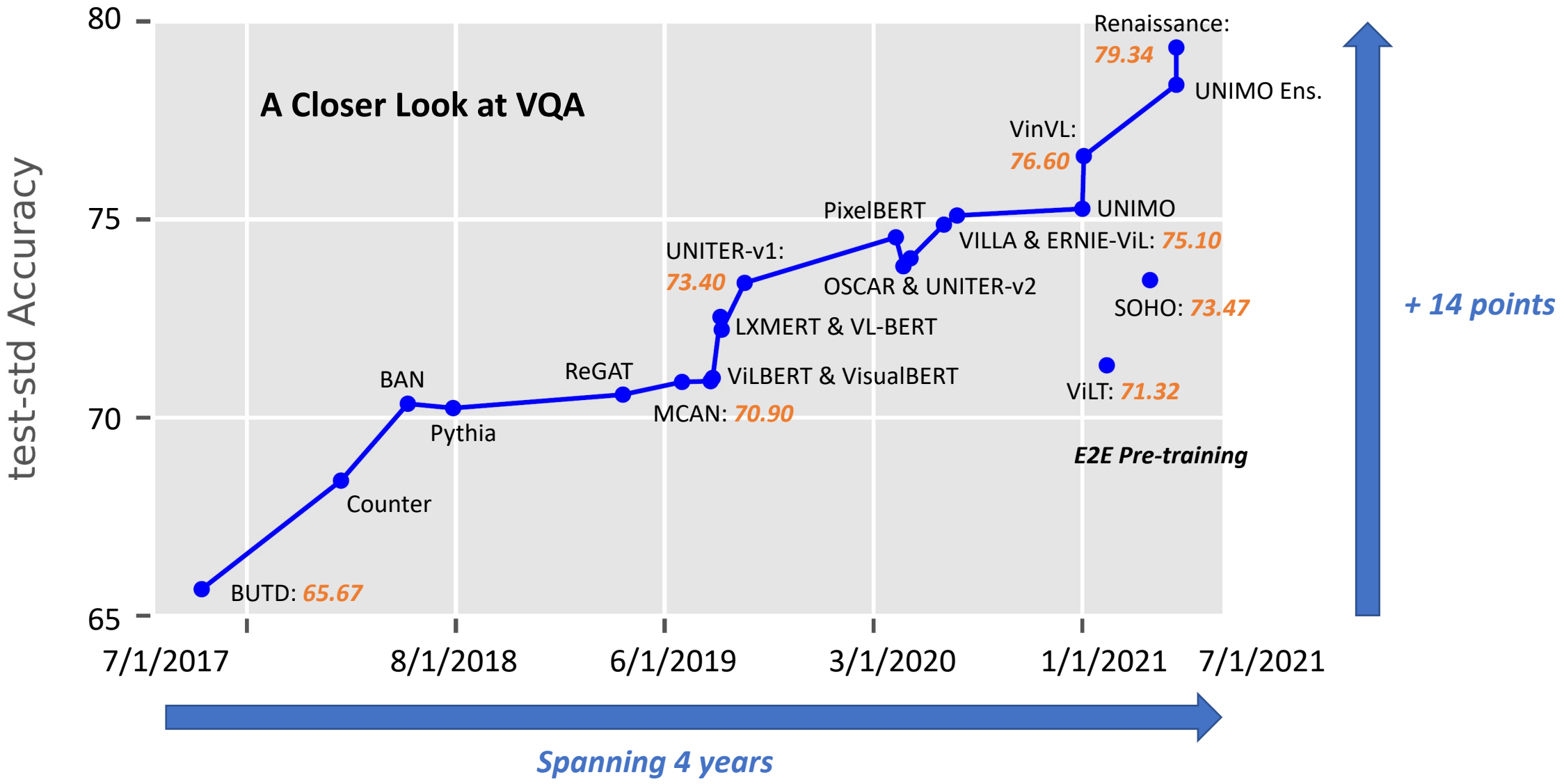

rug


chair


painting


plant

[1] ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision, ICML 2021


A Closer Look at VQA
test-std Accuracy
80
75
70
65
7/1/2017
8/1/2018
6/1/2019
3/1/2020
1/1/2021
7/1/2021
BUTD: 65.67
Counter
BAN
Pythia
ReGAT
MCAN: 70.90
ViLBERT & VisualBERT
LXMERT & VL-BERT
UNITER-v1: 73.40
OSCAR & UNITER-v2
PixelBERT
VILLA & ERNIE-ViL: 75.10
UNIMO
VinVL: 76.60
Renaissance: 79.34
UNIMO Ens.
SOHO: 73.47
ViLT: 71.32
E2E Pre-training
+ 14 points
Spanning 4 years

Challenges and Future Directions

- How can we close the performance gap between ViLT and VinVL?

- *Backbone architecture*:
  - Can simple MLPs such as MLP-Mixer also work well when we have enough training data to conquer model inductive bias?

- *Human parity*:
  - How far away do we from human parity on VQA?
  - How meaningful is it when we say human parity?
  - What are the future vision-language tasks?

Thank you!
Any Questions?