Vision Language Pretrainig

Pixel Bert / VinVL / ViLT

Georgia Tech

Kevin Rojas

- ML PhD Student at Math Department
- I’ve done work on multimodal diffusion models!
- Working on multimodal generative models for scientific applications!
- Looking for teammates!

Outline

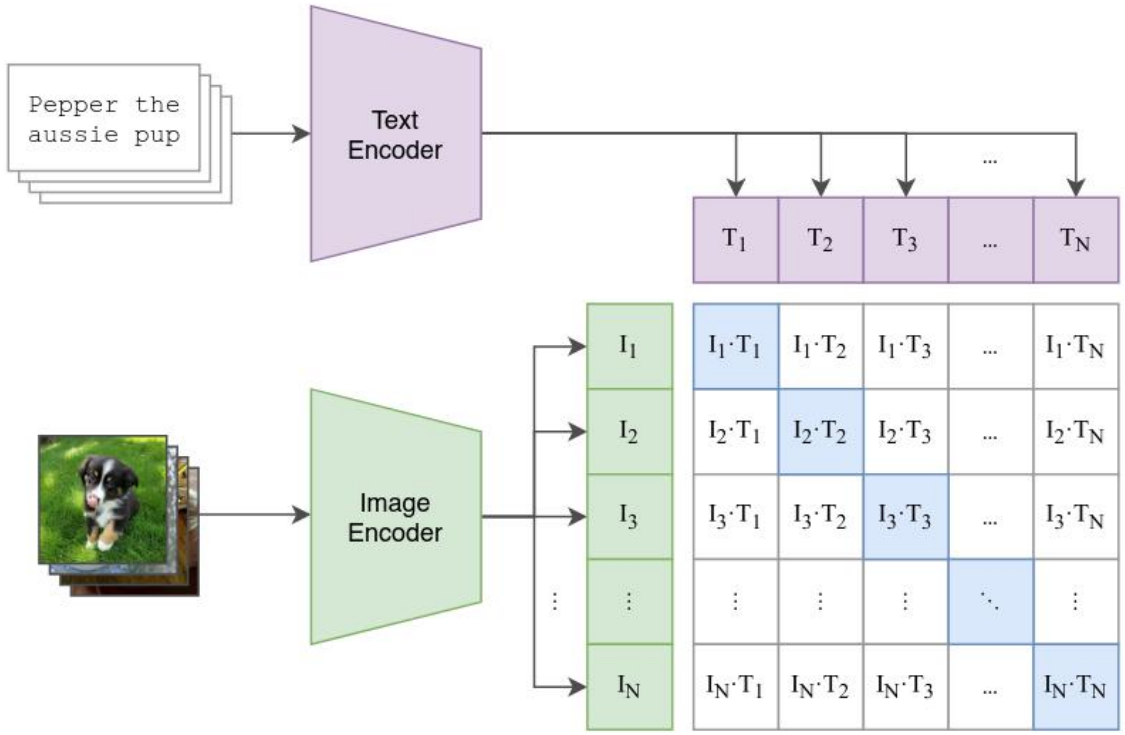
- Problem Statement
- Related Works
- Approach
- Experiments & Results
- Comparison

Problem Statement

- Vision Language Pretraining is key

- It requires
  - Text encoder
  - Vision encoder
  - Loss function

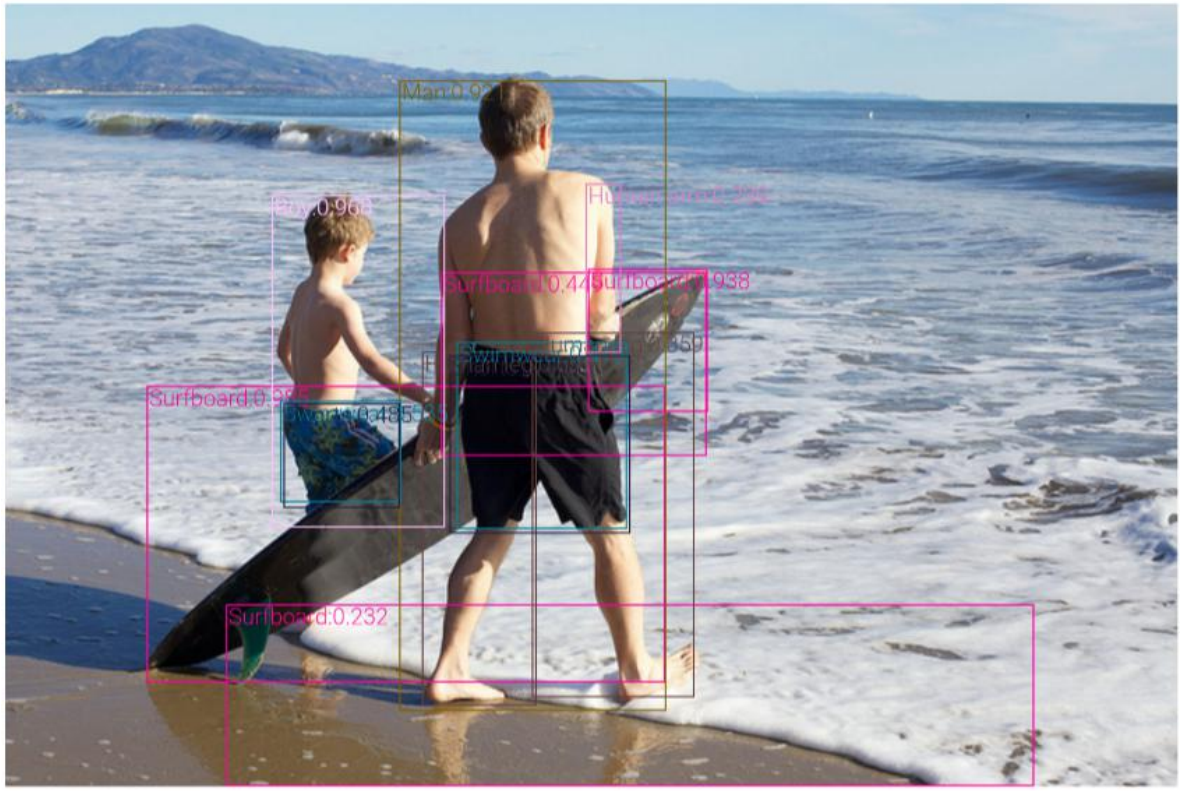
- With these papers we will study:
  - Vision encoder
  - Loss function



Georgia Tech

Problem Statement

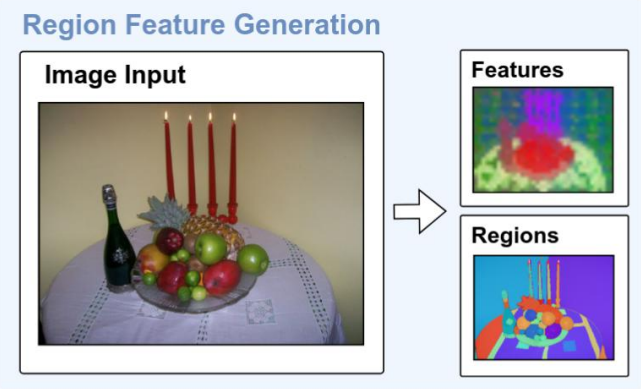
- Picking the visual representation is usually a big bottleneck
  - Region Based Features
  - Grid Features
  - Patch Projection



Georgia Tech

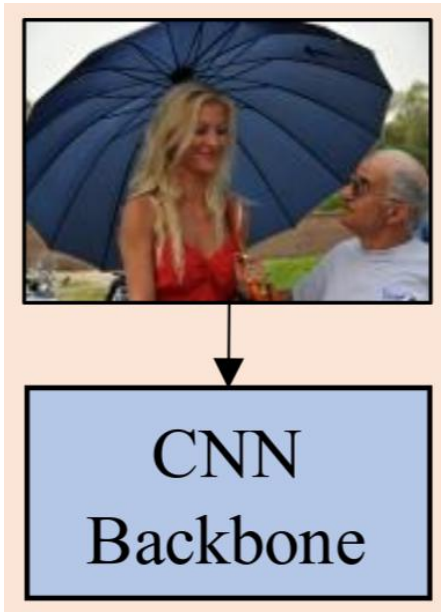
Three Approaches

Region Based Features



For instance as in last week

Pixel Level Features



Patch Level

Georgia Tech

Three Approaches

Pixel-BERT

- Object detection is limiting factor
- CNN
- Pixel level

VinVL

- Improve object detection
- CNN
- Region Feature

ViLT

- No object detection is needed
- Transformer
- Patch Embeddings

General VLM pipeline

1. Use a pretrained OD model to encode an image

2. Use a cross-modal fusion to align text + image

Pixel-BERT

Q: What is the plane doing?
A: Taking off

Example (A)

Q: Is the girl touching the ground?
A: No

Example (B)

Q: Is the animal moving?
A: Yes

Example (C)

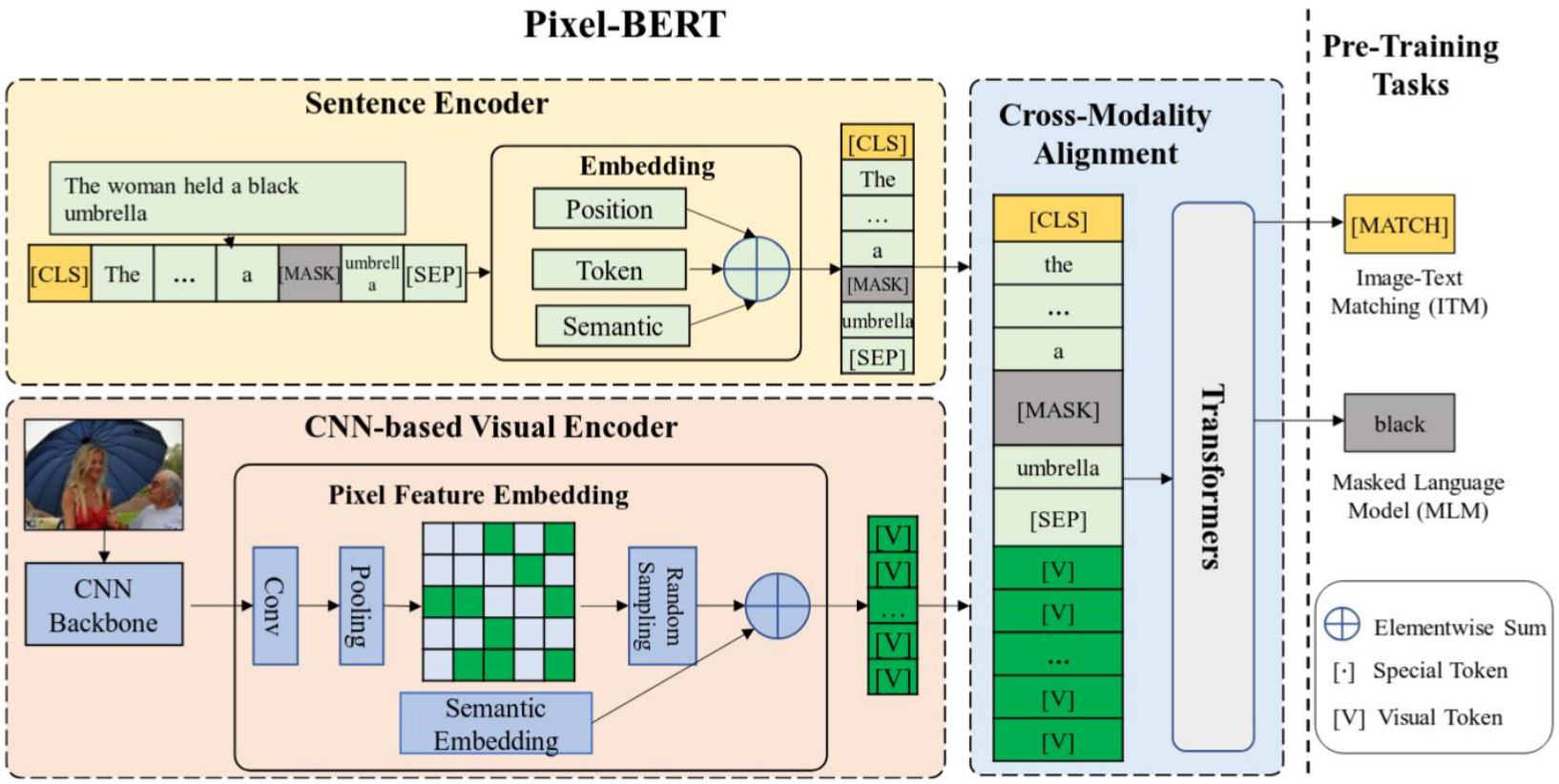
Pixel-BERT

- Region based features are designed for certain tasks (object detection)
- This leads to an information gap
- Bounding a region doesn't give language understanding!
- We need something else!

Q: What is the plane doing?
A: Taking off

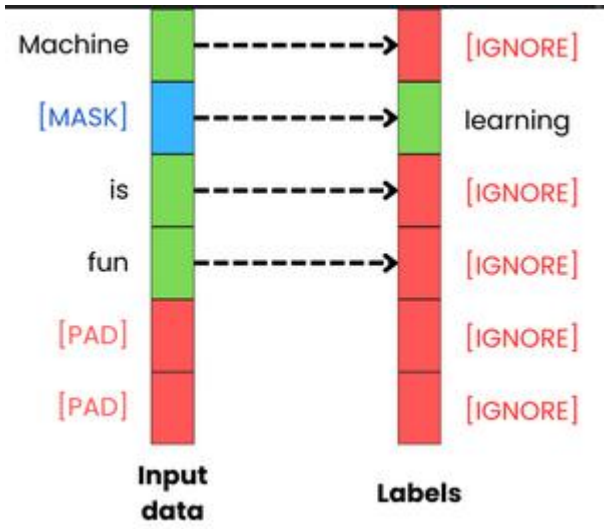
Example (A)

Pixel-BERT



Loss Functions

- Masked Language Modeling



$$\mathcal{L}_{\text{MLM}}(\theta) = -E_{(\mathbf{w},I)\sim D} \log P_\theta(w_m|\mathbf{w}_{\setminus m}, I),$$

- Image-Text Matching

Q: Is the animal moving?
A: Yes

$$\mathcal{L}_{\text{ITM}}(\theta) = -E_{(\mathbf{w},I)\sim D}[y \log S_\theta(\mathbf{w}, I) + (1-y)\log(1 - S_\theta(\mathbf{w}, I))],$$

General VLM pipeline

1. Use a pretrained OD model to encode an image

2. Use a cross-modal fusion to align text + image

- The OD model was treated as a black box
- A very old OD model was being used

Vin-VL

- If object detection is the bottleneck, lets fix it!
- Better model
- Better data

Data

- Make sure that we have at least 2000 samples per class for Objects 365/Open-Images
- Balanced every dataset (25% each)
- Merge vocabularies

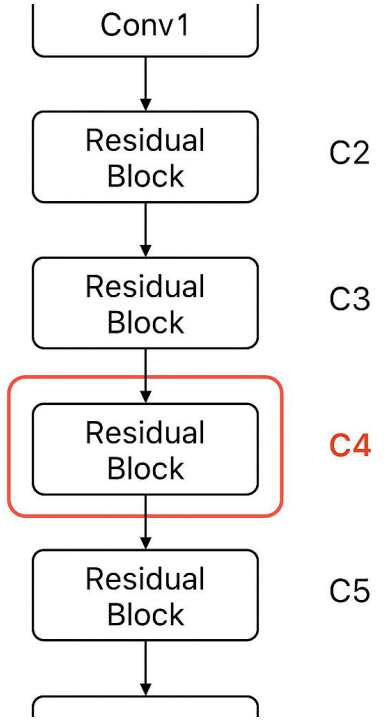
Source	VG	COCO w/ stuff	Objects365	OpenImagesV5	Total
Image	97k	111k	609k	1.67M	2.49M
classes	1594	171	365	500	1848
Sampling	×8	×8	CA-2k, ×2	CA-2k	5.43M

Model

- Feature Pyramid Network (FPN)

(d) Feature Pyramid Network

- Resnet C4



Georgia Tech

Model

- Feature Pyramid Network (FPN)
  - Outperforms C4 for object detection
- Resnet C4
  - Better visual features
  - The reason for the improvement is that C4 leverages pretraining better

Loss Functions

- The use a similar loss to pixel-bert

$$\mathcal{L}_{\text{Pre-training}} = \mathcal{L}_{\text{MTL}} + \mathcal{L}_{\text{CL3}}.$$

- Specifically the contrastive loss is:

$$\mathcal{L}_{\text{CL3}} = -\mathbb{E}_{(\boldsymbol{w},\boldsymbol{q},\boldsymbol{v};c)\sim\tilde{\mathcal{D}}} \log p(c|f(\boldsymbol{w},\boldsymbol{q},\boldsymbol{v})),$$

$$\boldsymbol{x} \triangleq (\underbrace{\boldsymbol{w}}_{\text{caption}}, \underbrace{\boldsymbol{q},\boldsymbol{v}}_{\text{tags\&image}}) \quad \text{or} \quad (\underbrace{\boldsymbol{w},\boldsymbol{q}}_{\text{Q\&A}}, \underbrace{\boldsymbol{v}}_{\text{image}})$$

- Where
  - c = 0 -----> triplet is matched
  - c = 1 -----> w is polluted
  - c = 2------> q is polluted

Ablations

- They perform ablations on Visual Question Answering (VQA)

- The model picks an answer from a set of options (3129)

Ablations

- The first ablation shows the effect of each pretraining

vision \ vl	no VLP	OSCAR$_B$ [21]	OSCAR+$_B$ (ours)
R101-C4 [2]	68.52 ±0.11	72.38	72.46±0.05
VinVL (ours)	71.34 ±0.17	–	74.90±0.05

Table 12: Effects of vision (V) and vision-language (VL) pre-training on VQA.

Ablations

- The second ablation shows the effect of data/model size

data \ model	R50-FPN	R50-C4	R101-C4 [2]	X152-C4
VG	67.35±0.26	67.86±0.31	68.52 ±0.11	69.10±0.06
4Sets→VG	68.3±0.11	68.39±0.16	–	71.34 ±0.17

Table 13: Ablation of model size and data size on training vision models.

Ablations

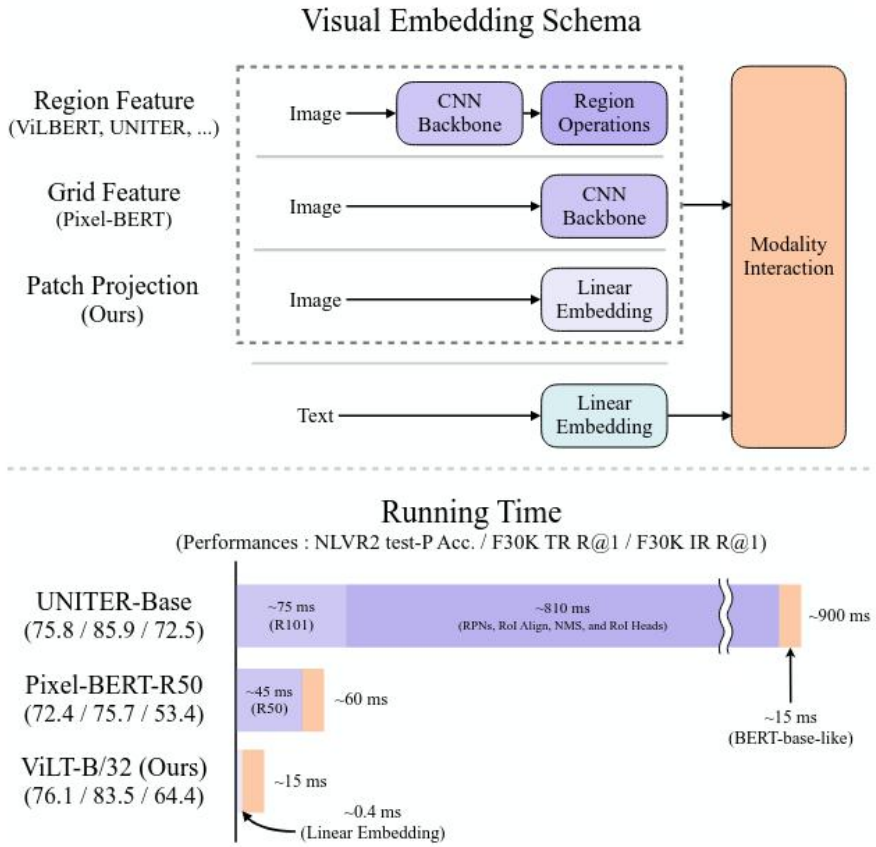
- The second ablation shows the effect of vocabulary size

Dataset name #obj & #attr	ImageNet 1000 & 0	VG-obj 317 & 0	VG w/o attr 1594 & 0	VG [2] 1600 & 400	VG 1594 & 524	4Sets→VG 1848 & 524
R50-C4 + BERT$_B$	66.13±0.04	64.25±0.16	66.51±0.11	67.63±0.25	67.86±0.31	68.39±0.16

Table 15: Effect of object-attribute vocabulary. We use all grid features (maximal 273) for the ImageNet classification model (first column), and maximal 50 region features for OD models (other columns).

Vi-LT

- Most VLP models use an object detector
- Pixel Bert is an exception
- Can we improve the visual embedders?



Georgia Tech

Taxonomy of VL models

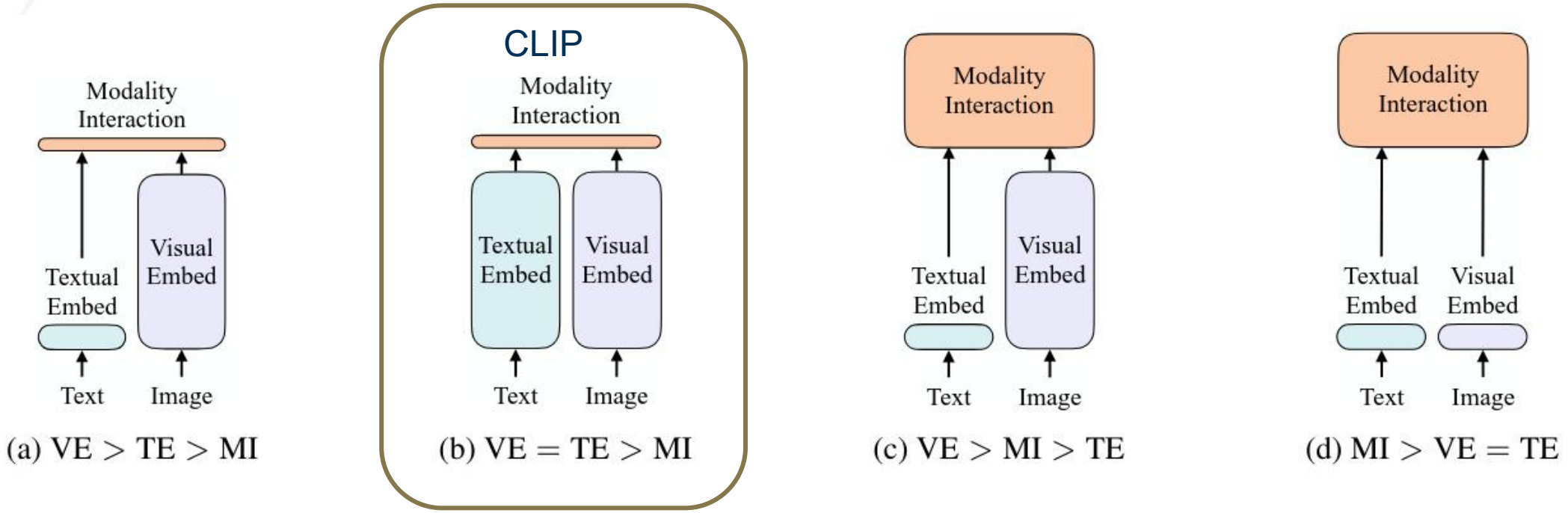


Figure 2. Four categories of vision-and-language models. The height of each rectangle denotes its relative computational size. VE, TE, and MI are short for visual embedder, textual embedder, and modality interaction, respectively.

Clip Limitations

- CLIP embeddings might not allow solving harder questions like NLVR2
- CLIP results in 50.99% accuracy
- Chance is 50%!
- The lack of **fusion** doesn't allow learning complex interactions

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

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

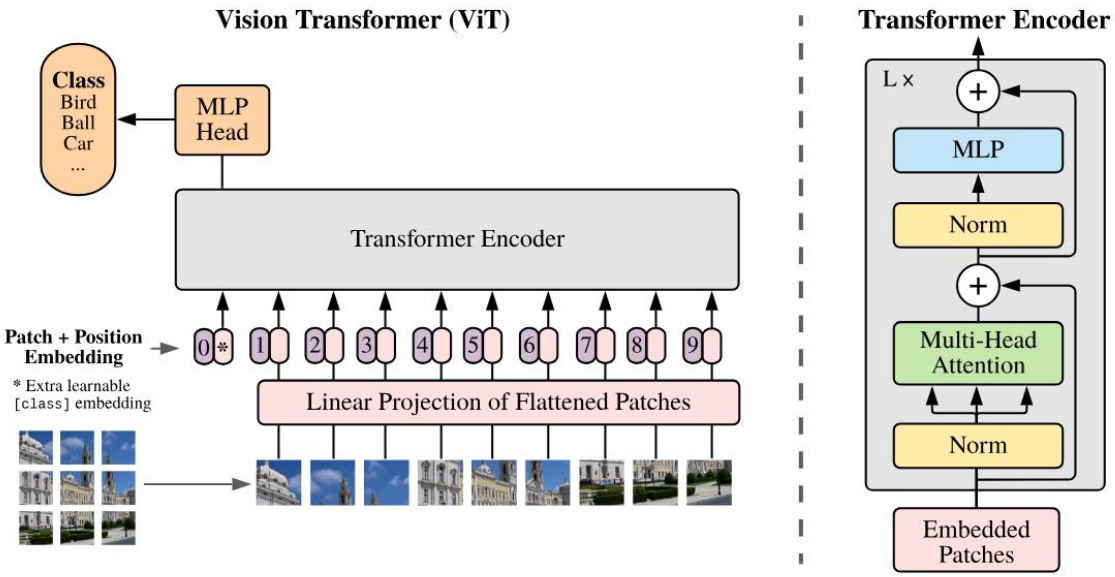
Visual Representations

- Region Feature
  - Vin-VL
- Expensive and complicated object detection pipelines

- Grid Features
  - Pixel-BERT
- CNNs can be expensive

- Patch Projection
  - Vi-LT
- Cheap and simple linear projection

ViT Model



$$\bar{t} = [t_{\text{class}}; t_1 T; \cdots; t_L T] + T^{\text{pos}} \tag{1}$$

$$\bar{v} = [v_{\text{class}}; v_1 V; \cdots; v_N V] + V^{\text{pos}} \tag{2}$$

$$z^0 = [\bar{t} + t^{\text{type}}; \bar{v} + v^{\text{type}}] \tag{3}$$

$$\hat{z}^d = \text{MSA}(\text{LN}(z^{d-1})) + z^{d-1}, \qquad d = 1 \ldots D \tag{4}$$

$$z^d = \text{MLP}(\text{LN}(\hat{z}^d)) + \hat{z}^d, \qquad d = 1 \ldots D \tag{5}$$

$$p = \tanh(z_0^D W_{\text{pool}}) \tag{6}$$

Their Model

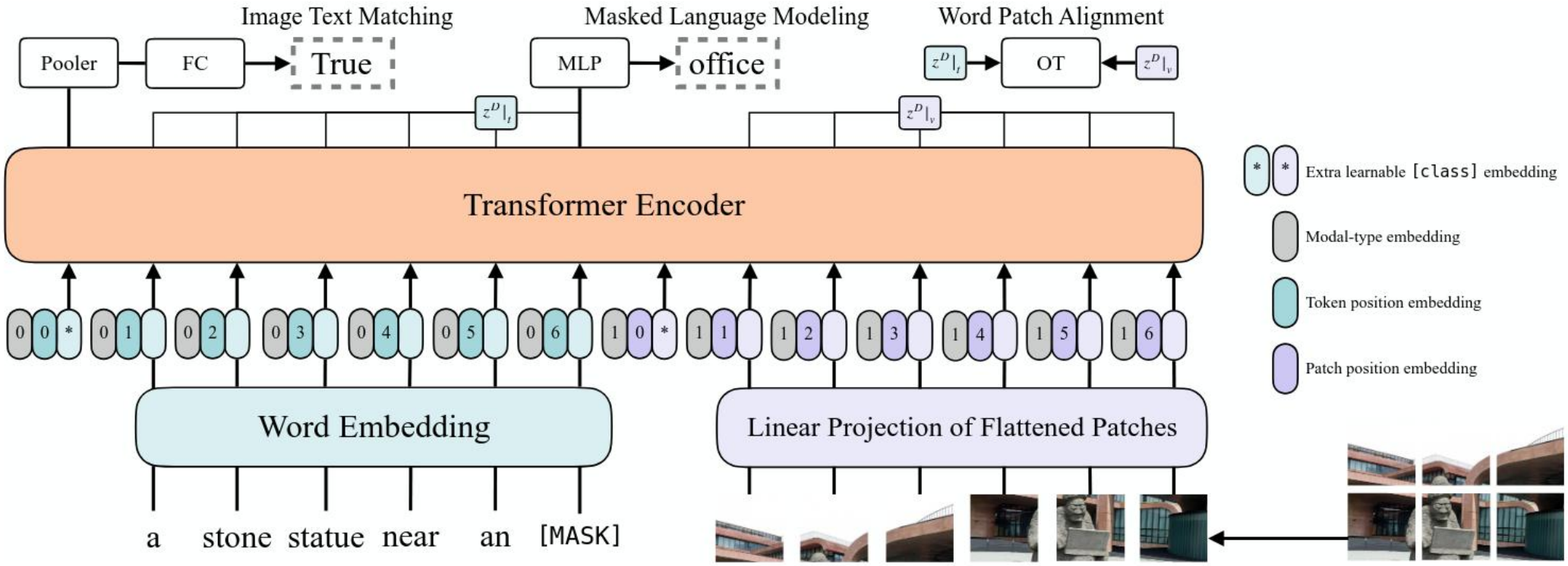
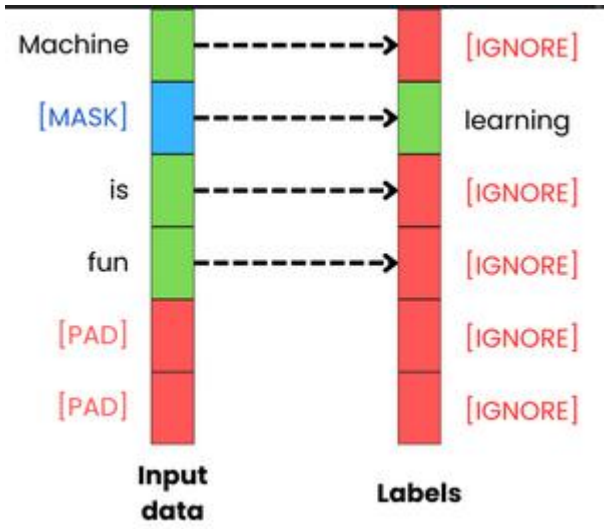


Figure 3. Model overview. Illustration inspired by Dosovitskiy et al. (2020).

Loss Functions

- Masked Language Modeling



$$\mathcal{L}_{\text{MLM}}(\theta) = -E_{(\mathbf{w},I)\sim D} \log P_\theta(w_m|\mathbf{w}_{\setminus m}, I),$$

- Image-Text Matching

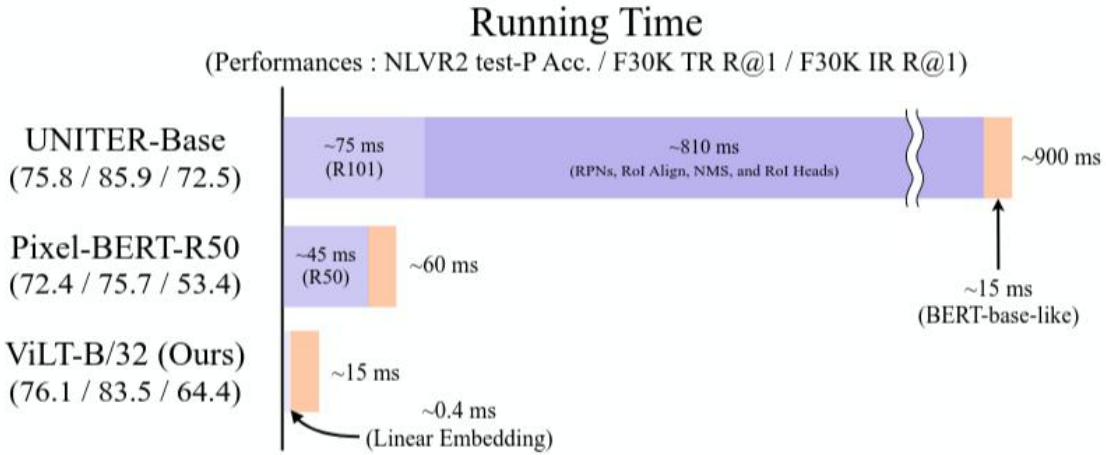
Q: Is the animal moving?
A: Yes

$$\mathcal{L}_{\text{ITM}}(\theta) = -E_{(\mathbf{w},I)\sim D}[y \log S_\theta(\mathbf{w}, I) + (1-y)\log(1 - S_\theta(\mathbf{w}, I))],$$

Experiments and Results

- Question Answering

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	~925	70.80	67.40	67.00
	LXMERT	~900	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†‡	~650	75.95	82.05	83.08
Grid	Pixel-BERT-X152	~160	74.45	76.50	77.20
	Pixel-BERT-R50	~60	71.35	71.70	72.40
Linear	ViLT-B/32	~15	70.33	74.41	74.57
	ViLT-B/32ⓐ	~15	70.85	74.91	75.57
	ViLT-B/32ⓐ⊕	~15	71.26	75.70	76.13



Georgia Tech

• Question Answering

Takeaway

- Using a linear projection can result in faster computations
- For question answering Linear projection are as competitive as Grid based methods like Pixel-BERT

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	~925	70.80	67.40	67.00
	LXMERT	~900	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†‡	~650	75.95	82.05	83.08
Grid	Pixel-BERT-X152	~160	74.45	76.50	77.20
	Pixel-BERT-R50	~60	71.35	71.70	72.40
Linear	ViLT-B/32	~15	70.33	74.41	74.57
	ViLT-B/32ⓐ	~15	70.85	74.91	75.57
	ViLT-B/32ⓐ⊕	~15	71.26	75.70	76.13

Experiments and Results

- Retrieval Tasks

Table 4. Comparison of ViLT-B/32 with other models on downstream retrieval tasks. We use SCAN for w/o VLP SOTA results. † additionally used GQA, VQAv2, VG-QA for pre-training. ‡ additionally used the Open Images dataset. ⓐ indicates RandAugment is applied during fine-tuning. ⊕ indicates model trained for a longer 200K pre-training steps.

Visual Embed	Model	Time (ms)	Text Retrieval						Image Retrieval					
			Flickr30k (1K)			MSCOCO (5K)			Flickr30k (1K)			MSCOCO (5K)		
			R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Region	w/o VLP SOTA	~900	67.4	90.3	95.8	50.4	82.2	90.0	48.6	77.7	85.2	38.6	69.3	80.4
	ViLBERT-Base	~920	-	-	-	-	-	-	58.2	84.9	91.5	-	-	-
	Unicoder-VL	~925	86.2	96.3	99.0	62.3	87.1	92.8	71.5	91.2	95.2	48.4	76.7	85.9
	UNITER-Base	~900	85.9	97.1	98.8	64.4	87.4	93.1	72.5	92.4	96.1	50.3	78.5	87.2
	OSCAR-Base†	~900	-	-	-	70.0	91.1	95.5	-	-	-	54.0	80.8	88.5
	VinVL-Base†‡	~650	-	-	-	74.6	92.6	96.3	-	-	-	58.1	83.2	90.1
Grid	Pixel-BERT-X152	~160	87.0	98.9	99.5	63.6	87.5	93.6	71.5	92.1	95.8	50.1	77.6	86.2
	Pixel-BERT-R50	~60	75.7	94.7	97.1	59.8	85.5	91.6	53.4	80.4	88.5	41.1	69.7	80.5
Linear	ViLT-B/32	~15	81.4	95.6	97.6	61.8	86.2	92.6	61.9	86.8	92.8	41.3	72.0	82.5
	ViLT-B/32ⓐ	~15	83.7	97.2	98.1	62.9	87.1	92.7	62.2	87.6	93.2	42.6	72.8	83.4
	ViLT-B/32ⓐ⊕	~15	83.5	96.7	98.6	61.5	86.3	92.7	64.4	88.7	93.8	42.7	72.9	83.1

• Retrieval Tasks

Takeaway

- Using a linear projection can result in faster computations
- The pretrained embeddings from Grid/Region based visual encoders tend to produce better results

Table 4. Comparison of ViLT-B/32 with other models on downstream retrieval tasks. We use SCAN for w/o VLP SOTA results. † additionally used GQA, VQAv2, VG-QA for pre-training. ‡ additionally used the Open Images dataset. ⓐ indicates RandAugment is applied during fine-tuning. ⊕ indicates model trained for a longer 200K pre-training steps.

Visual Embed	Model	Time (ms)	Text Retrieval						Image Retrieval					
			Flickr30k (1K)			MSCOCO (5K)			Flickr30k (1K)			MSCOCO (5K)		
			R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
Region	w/o VLP SOTA	~900	67.4	90.3	95.8	50.4	82.2	90.0	48.6	77.7	85.2	38.6	69.3	80.4
	ViLBERT-Base	~920	-	-	-	-	-	-	58.2	84.9	91.5	-	-	-
	Unicoder-VL	~925	86.2	96.3	99.0	62.3	87.1	92.8	71.5	91.2	95.2	48.4	76.7	85.9
	UNITER-Base	~900	85.9	97.1	98.8	64.4	87.4	93.1	72.5	92.4	96.1	50.3	78.5	87.2
	OSCAR-Base†	~900	-	-	-	70.0	91.1	95.5	-	-	-	54.0	80.8	88.5
	VinVL-Base†‡	~650	-	-	-	74.6	92.6	96.3	-	-	-	58.1	83.2	90.1
Grid	Pixel-BERT-X152	~160	87.0	98.9	99.5	63.6	87.5	93.6	71.5	92.1	95.8	50.1	77.6	86.2
	Pixel-BERT-R50	~60	75.7	94.7	97.1	59.8	85.5	91.6	53.4	80.4	88.5	41.1	69.7	80.5
Linear	ViLT-B/32	~15	81.4	95.6	97.6	61.8	86.2	92.6	61.9	86.8	92.8	41.3	72.0	82.5
	ViLT-B/32ⓐ	~15	83.7	97.2	98.1	62.9	87.1	92.7	62.2	87.6	93.2	42.6	72.8	83.4
	ViLT-B/32ⓐ⊕	~15	83.5	96.7	98.6	61.5	86.3	92.7	64.4	88.7	93.8	42.7	72.9	83.1

Other Engineering Techniques

- **Image Augmentation**

- Apply random changes to images to get “more data” using RandAugment

Geometric Transforms

- Shear X
- Shear Y
-Translate X
- Translate Y
- Rotate

Color Transformations

- AutoContrast
- Invert
-Equalize
-Solarize
- Contrast
- Color
-Brightness
-Shapness

- **Whole Word Masking**

- Mask the entire word not only some of its tokens

- Giraffe -> [gi , raf , fe]

Bad

[gi, [Mask], fe]

Good

[[Mask], [Mask], [Mask]]

Other Engineering Techniques

Table 5. Ablation study of ViLT-B/32. ⓦ denotes whether whole word masking is used for pre-training. ⓜ denotes whether MPP objective is used for pre-training. ⓐ denotes whether RandAugment is used during fine-tuning.

Training Steps	Ablation ⓦ	ⓜ	ⓐ	VQAv2 test-dev	NLVR2 dev	test-P	Flickr30k R@1 (1K) TR (ZS)	IR (ZS)	MSCOCO R@1 (5K) TR (ZS)	IR (ZS)
25K	X	X	X	68.96 ± 0.07	70.83 ± 0.19	70.83 ± 0.23	75.39 (45.12)	52.52 (31.80)	53.72 (31.55)	34.88 (21.58)
50K	X	X	X	69.80 ± 0.01	71.93 ± 0.27	72.92 ± 0.82	78.13 (55.57)	57.36 (40.94)	57.00 (39.56)	37.47 (27.51)
100K	X	X	X	70.16 ± 0.01	73.54 ± 0.02	74.15 ± 0.27	79.39 (66.99)	60.50 (47.62)	60.15 (51.25)	40.45 (34.59)
100K	O	X	X	70.33 ± 0.01	74.41 ± 0.21	74.57 ± 0.09	81.35 (69.73)	61.86 (51.28)	61.79 (53.40)	41.25 (37.26)
100K	O	O	X	70.21 ± 0.05	72.76 ± 0.50	73.54 ± 0.47	78.91 (63.67)	58.76 (46.96)	59.53 (47.75)	40.08 (32.28)
100K	O	X	O	70.85 ± 0.13	74.91 ± 0.29	75.57 ± 0.61	83.69 (69.73)	62.22 (51.28)	62.88 (53.40)	42.62 (37.26)
200K	O	X	O	71.26 ± 0.06	75.70 ± 0.32	76.13 ± 0.39	83.50 (73.24)	64.36 (54.96)	61.49 (56.51)	42.70 (40.42)

Other Engineering Techniques

- Applying full word masking is beneficial

- Applying data augmentations is beneficial

Table 5. Ablation study of ViLT-B/32. ⓦ denotes whether whole word masking is used for pre-training. ⓜ denotes whether MPP objective is used for pre-training. ⓐ denotes whether RandAugment is used during fine-tuning.

Training Steps	Ablation ⓦ	ⓜ	ⓐ	VQAv2 test-dev	NLVR2 dev	test-P	Flickr30k R@1 (1K) TR (ZS)	IR (ZS)	MSCOCO R@1 (5K) TR (ZS)	IR (ZS)
25K	X	X	X	68.96 ± 0.07	70.83 ± 0.19	70.83 ± 0.23	75.39 (45.12)	52.52 (31.80)	53.72 (31.55)	34.88 (21.58)
50K	X	X	X	69.80 ± 0.01	71.93 ± 0.27	72.92 ± 0.82	78.13 (55.57)	57.36 (40.94)	57.00 (39.56)	37.47 (27.51)
100K	X	X	X	70.16 ± 0.01	73.54 ± 0.02	74.15 ± 0.27	79.39 (66.99)	60.50 (47.62)	60.15 (51.25)	40.45 (34.59)
100K	O	X	X	70.33 ± 0.01	74.41 ± 0.21	74.57 ± 0.09	81.35 (69.73)	61.86 (51.28)	61.79 (53.40)	41.25 (37.26)
100K	O	O	X	70.21 ± 0.05	72.76 ± 0.50	73.54 ± 0.47	78.91 (63.67)	58.76 (46.96)	59.53 (47.75)	40.08 (32.28)
100K	O	X	O	70.85 ± 0.13	74.91 ± 0.29	75.57 ± 0.61	83.69 (69.73)	62.22 (51.28)	62.88 (53.40)	42.62 (37.26)
200K	O	X	O	71.26 ± 0.06	75.70 ± 0.32	76.13 ± 0.39	83.50 (73.24)	64.36 (54.96)	61.49 (56.51)	42.70 (40.42)

Three Approaches Comparison

Pixel-BERT

- CNN
- Pixel level
- Good Embeddings
- Middle Ground

VinVL

- CNN
- Region Feature
- Best embeddings
- Slow

ViLT

- Transformer
- Patch Embeddings
- Worse embeddings
- Fast

Thank you!

Georgia Tech