ViLT : Vision-and-Language Transformer Without Convolution or Region Supervision

School of Industrial and Management Engineering, Korea University

Sae Rin Lim

KOREA UNIVERSITY

DMQA

• Contents

• Introduction

❖ ViLT : Vision-and-Language Transformer Without Convolution or Region Supervision

- 2021년 10월 30일 기준 29회 인용
- ICML 2021에 Long Talk으로 게재 승인
- 카카오브레인, 카카오엔터프라이즈, 카카오가 함께한 공동연구
- 기존 VLP 모델의 연산 병목현상 및 깊은 구조를 간단하고 효율적으로 개선하는 모델 제안

ViLT: Vision-and-Language Transformer Without Convolution or Region Supervision

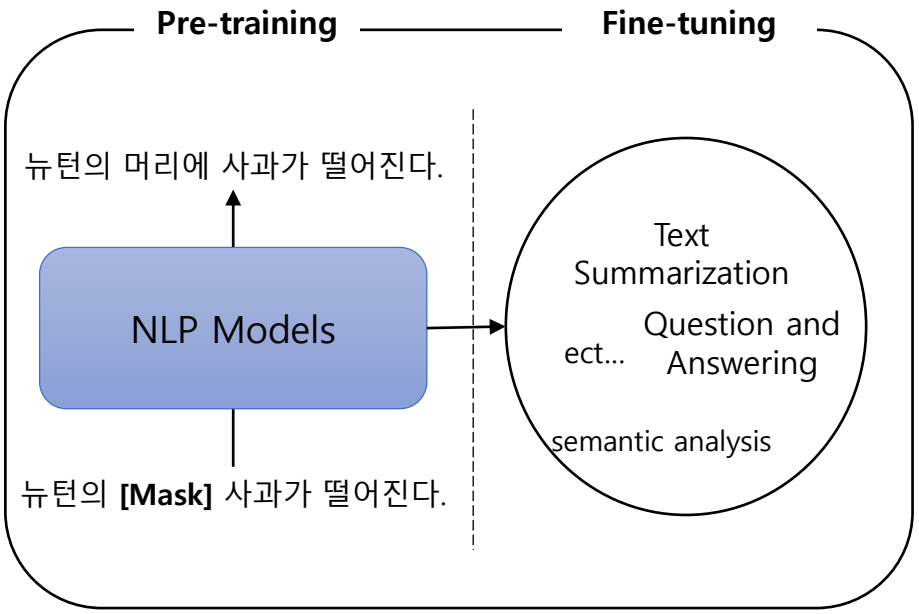
---

Wonjae Kim [*1 †] **Bokyung Son** [*1] **Ildoo Kim** [2]

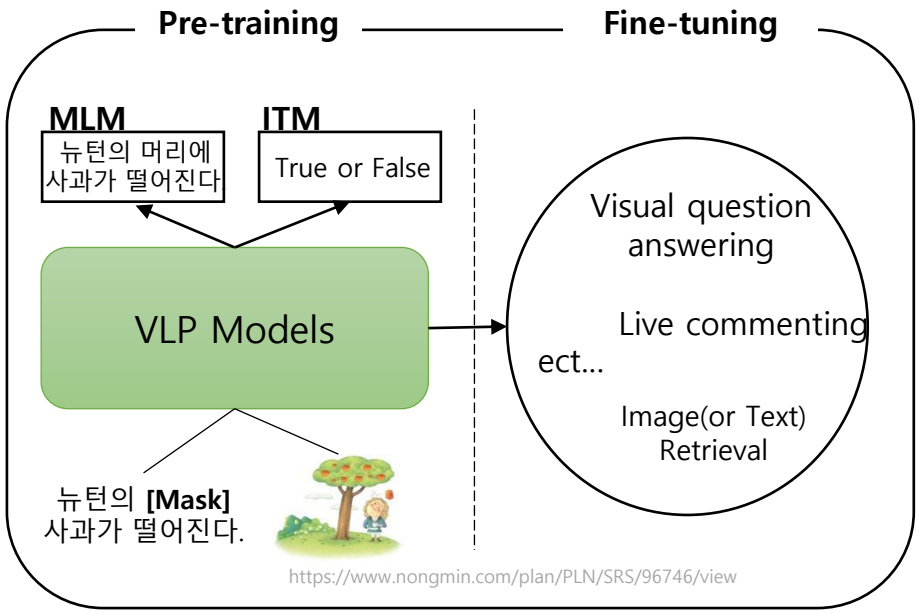
• Background

❖ VLP : Vision and Language Pre-training models

- 이미지가 함께 이미지를 설명하는 텍스트가 주어지는 **NLP 모델의 확장 버전**
- 자가지도학습을 통해 이미지와 텍스트가 동시에 주어지는 **다양한 과업들을 적은 레이블로도 해결**하는 것이 목표
- 일반적으로 **Masked Language Modeling(MLM)**과 동시에 **Image Text Matching(ITM)** 문제를 사전학습 과업으로 활용



Natural Language Models



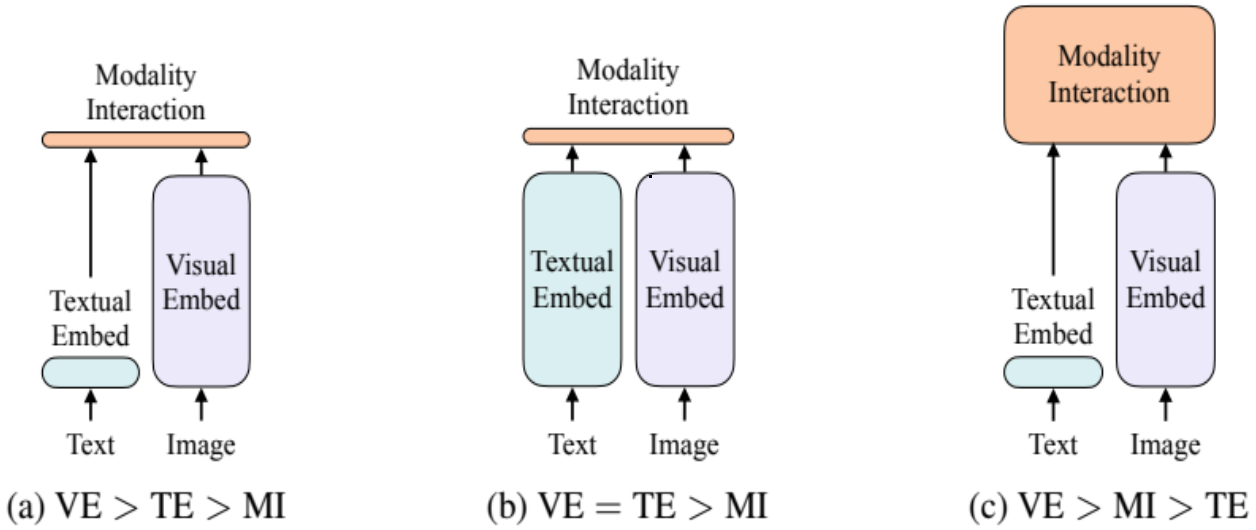
Vision and Language Pre-training Models

DMQA

• Background

❖ Previous Study of VLP Models

- 기존 연구되었던 VLP 모델은 모듈의 연산 속도 별로 크게 3가지로 나눌 수 있음
- 지금까지의 연구에서는 Visual Embedder로 Deep CNN을 사용
- 특히 Region feature를 사용하는 모델이 지배적이기 때문에 Object detection 과정이 추가적으로 필요



기존 VLP 모델 구조 (두께는 연산량을 나타냄)
VE : Visual Embedder, TE : Textual Embedder, MI : Modality Interaction

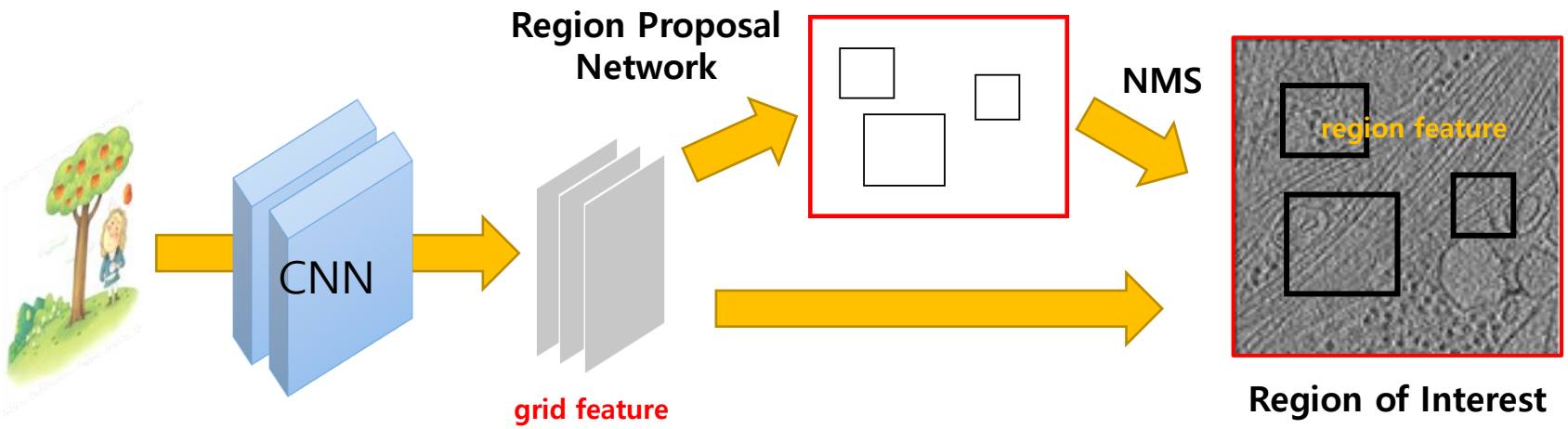
DMQA

• Background

❖ Visual Embedding Schema

- 기존 Visual Embedding **region feature embedding** 혹은 **grid feature embedding** 으로 나뉨
- Region feature는 Region of Interest(RoI)에 대한 feature로 **중요 pixel에 대한 정보**를 가지나 Region Proposal Net과 NMS 과정이 추가로 필요, 이전 연구의 대부분 VLP 모델에서 사용
- Grid feature는 CNN에서 출력된 feature map으로 이미지 전체의 pixel 정보를 가지고 있음

* NMS : Non Maximum Suppression, 중복된 RoI를 제거하는 알고리즘



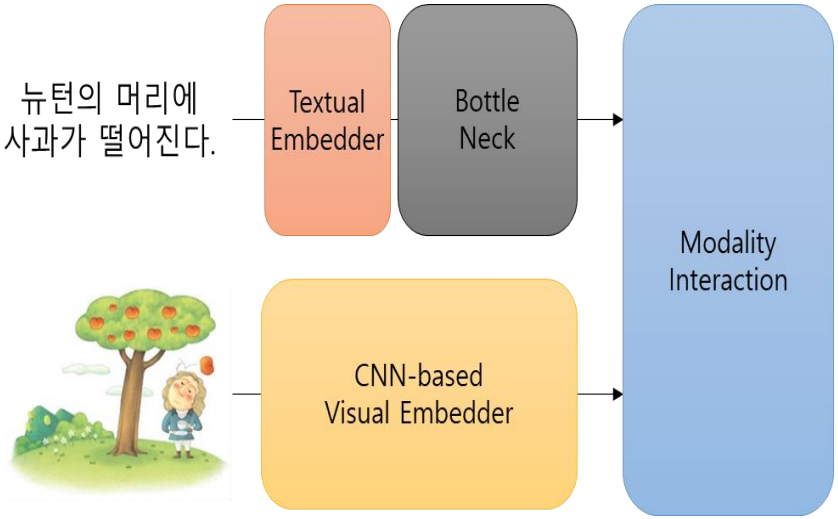
CNN-based Visual Embedder Process

DMQA

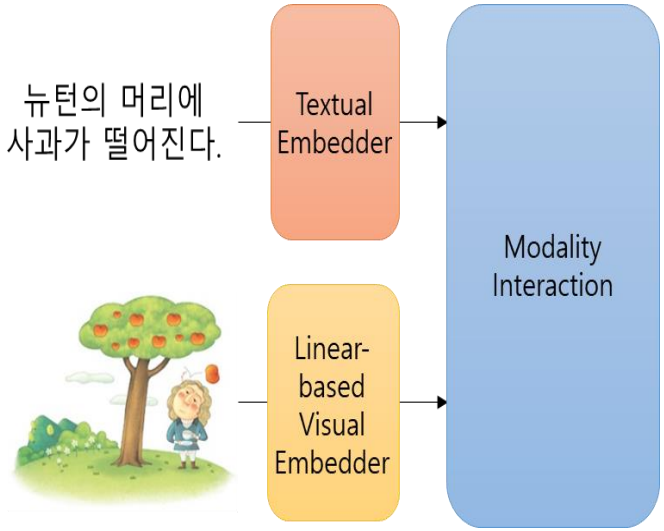
• ViLT : Vision-and Language Transformer

❖ Problems and Main Idea

- CNN-based Visual Embedder와 Linear-based Textual Embedder의 **연산 속도 차이로 인해 병목현상 발생**
- Deep CNN으로 Visual Embedder를 구성하기 때문에 모델 자체가 무거워 **많은 컴퓨터 자원**을 필요로 함
- 본 논문에서는 해당 문제를 해결하기 위해 ViT처럼 **Linear embedder만을 활용**하여 약 10배 빠른 모델 제안
- Lenear embedder를 통해 출력된 grid feature를 사용하기 때문에 추가적인 detection 과정이 없음



기본적인 VLP 모델



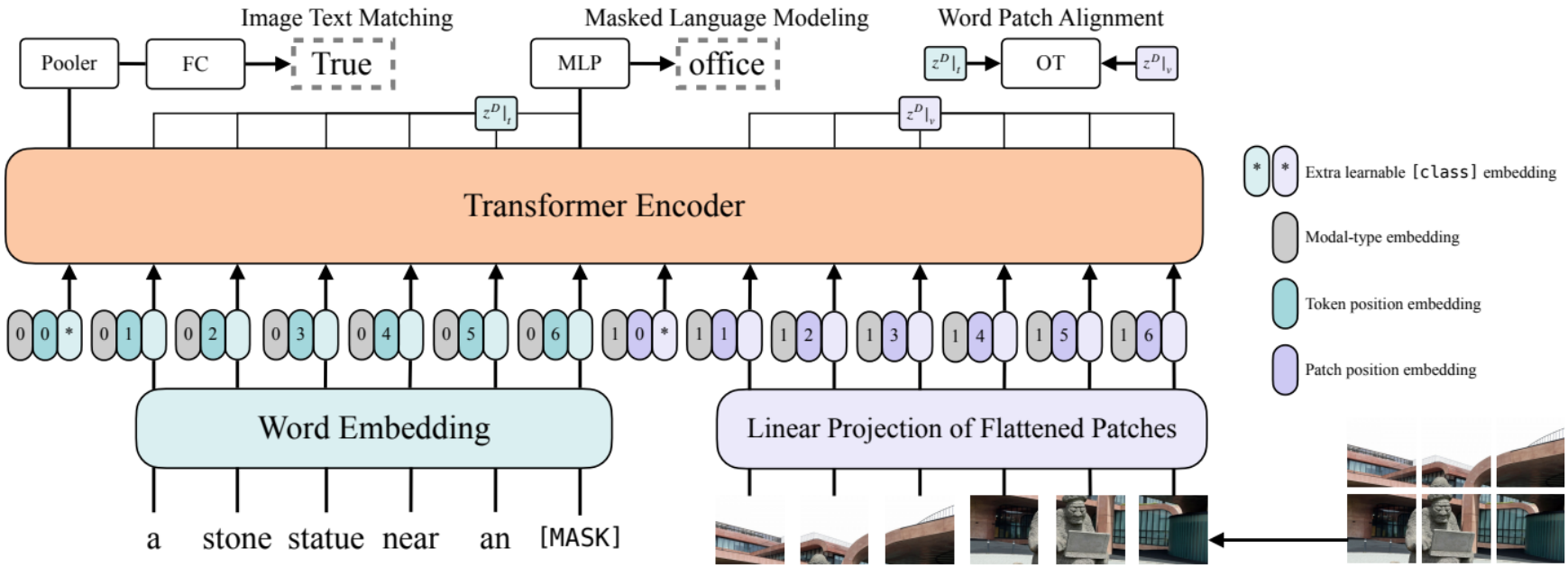
제안 VLP 모델

DMQA

• ViLT : Vision-and Language Transformer

❖ Architecture

- Text는 기존 Transformer와 같이 토큰화 후 Word Embedding을 진행하여 입력
- Image는 기존 ViT와 같이 Patch단위로 토큰화 후 Linear Projection을 진행하여 입력
- Image Text Matching을 위해 학습가능한 Class Token 추가



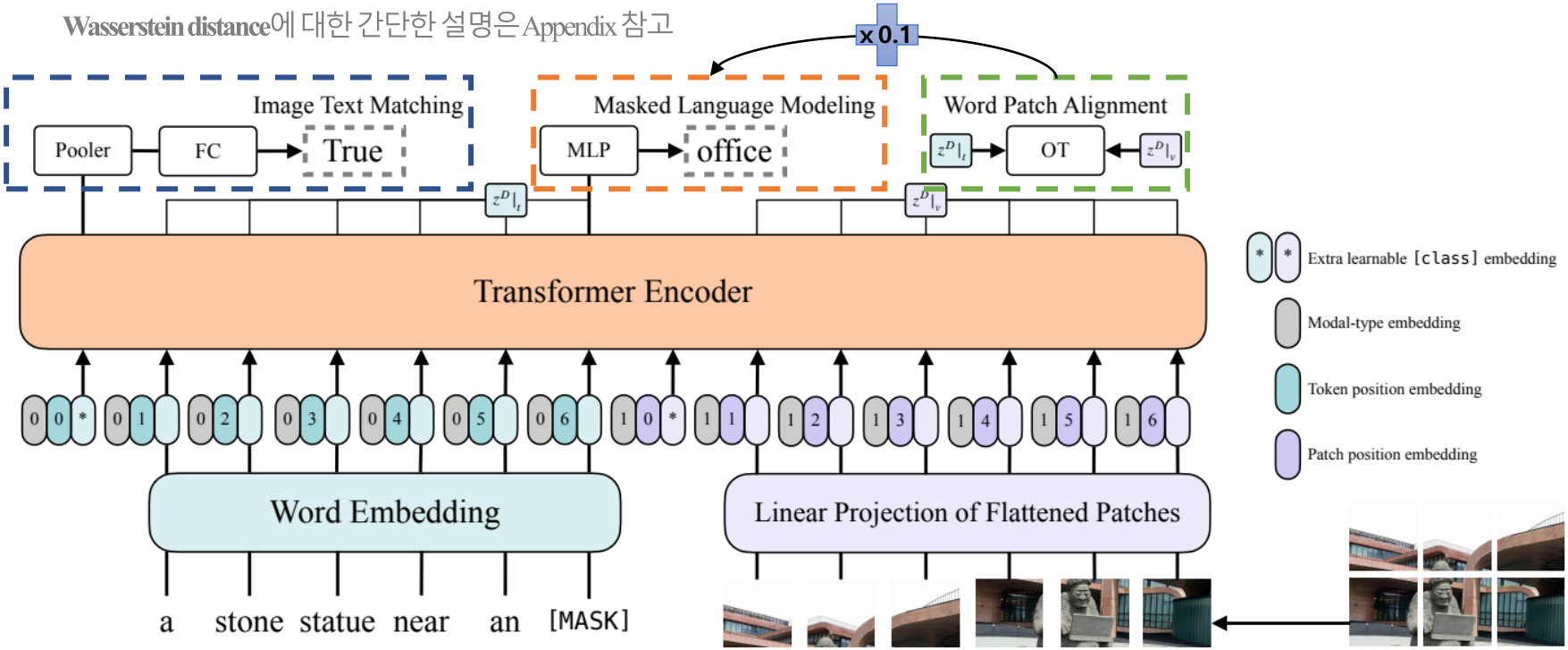
ViLT Model overview

• ViLT : Vision-and Language Transformer

❖ Pre-training Objectives

- 기존 연구들과 마찬가지로 **Masked Language Modeling**과 **Image Text Matching**을 활용
- 저자들은 Image Text Matching에 **Word Patch Alignment** 과업을 추가하여 0.1을 곱해 ITM Loss term에 추가
- Word Patch Alignment는 text와 image의 여러 부분집합 간의 attention score를 이용해 wasserstein distance를 추정하여 사용

Wasserstein distance에 대한 간단한 설명은 Appendix 참고



ViLT Model overview

DMQA

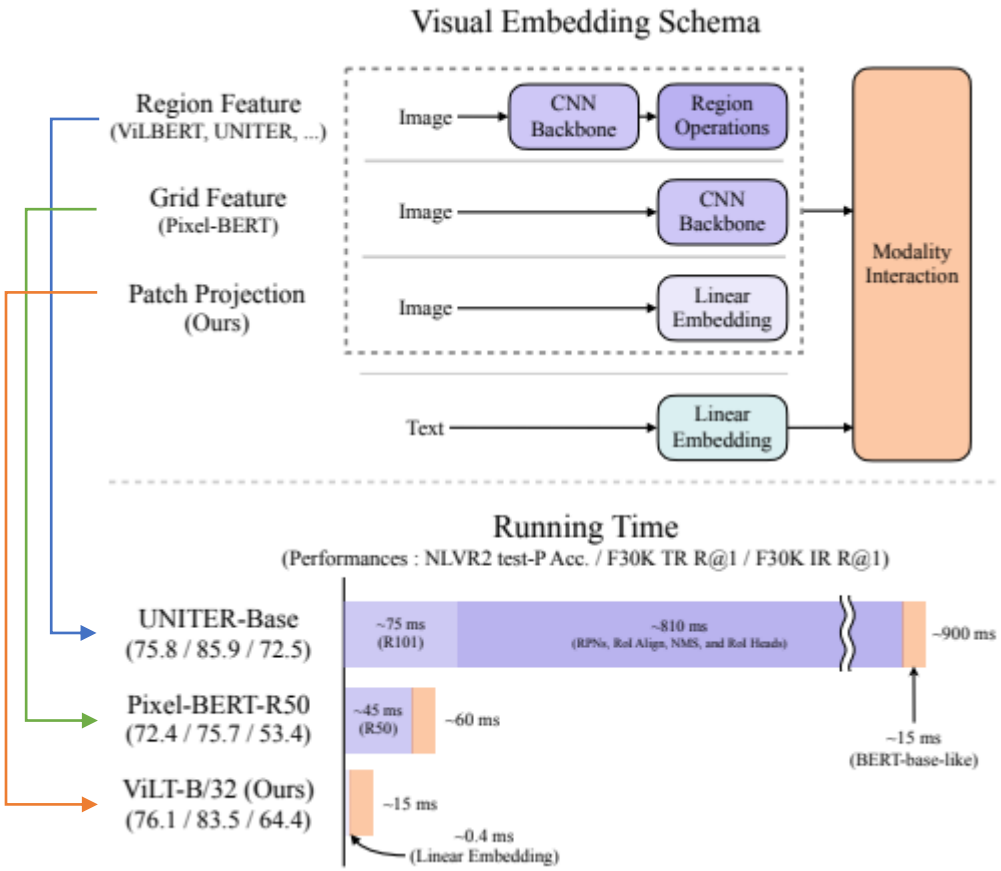
• Experiments

❖ Evaluation Datasets

Pre-training				Fine-tuning		
Dataset	Images	Captions	Caption Length	Dataset		Example
Microsoft COCO	113K	567K	11.81 ± 2.81	Classification	VQAv2	
Visual Genome	108K	5.41M	5.53 ± 1.76		NLVR2	
SBU Captions	867K	867K	10.66 ± 4.93	Retrieval	Microsoft COCO	
Google Conceptual Captions	3.01M	3.01M	15.0 ± 7.74		Flickr30K	

• Experiments

❖ Running Time Comparison



Visual Embedding 방법 별 Running Time 비교

Table 6. Comparison of VLP models in terms of parameter size, FLOPs, and inference latency. Since FLOPs are proportional to input size, we denote the number of input tokens (image+text) as superscripts ("?" when text length is unreported; we arbitrarily use length 40). Although not captured in FLOPs count nor parameter size (because it is not a tensor operation), note that *per-class* NMS for 1,600 classes amounts to more than 500 ms in latency. NMS latency varies a lot according to the number of detected classes.

Visual Embed	Model	#Params (M)	#FLOPs (G)	Time (ms)
Region	ViLBERT^{36+36}	274.3	958.1	~900
	VisualBERT^{36+128}	170.3	425.0	~925
	LXMERT^{36+20}	239.8	952.0	~900
	UNITER-Base^{36+60}	154.7	949.9	~900
	OSCAR-Base^{50+35}	154.7	956.4	~900
	VinVL-Base^{50+35}	157.3	1023.3	~650
	Unicoder-VL$^{100+?}$	170.3	419.7	~925
	ImageBERT^{100+44}	170.3	420.6	~925
Grid	Pixel-BERT-X152$^{146+?}$	144.3	185.8	~160
	Pixel-BERT-R50$^{260+?}$	94.9	136.8	~60
Linear	ViLT-B/32^{200+40}	87.4	55.9	~15

모델 별 FLOPs 및 Running Time 비교

• Experiments

❖ Downstream : Classification Task

- 기존 모델들에 비해 Time이 큰 폭으로 감소하였고 같은 크기의 pre-training dataset을 사용했을 때 성능이 비슷함

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

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 pre-training. ‡ made additional use of the Open Images (Kuznetsova et al., 2020) dataset. ⓐ indicates RandAugment is applied during fine-tuning. ⊕ indicates model trained for a longer 200K pre-training steps.

• Experiments

❖ Downstream : Retrieval Tasks (Zero-shot and Fine-tuning)

- 저자들은 VLP 모델 학습에서 처음으로 **Augmentation의 효과를 실험적으로 입증**, 실험에서는 RnadAugment를 사용

Zero-shot

Table 3. Comparison of ViLT-B/32 with other VLP models on downstream zero-shot retrieval tasks. We exclude the models of which zero-shot retrieval performances were not reported in their original papers. † is pre-trained with a 10M proprietary vision-and-language dataset in addition to the 4M dataset of GCC+SBU. ⊕ indicates model trained for a longer 200K pre-training steps.

Visual Embed	Model	Time (ms)	Zero-Shot Text Retrieval						Zero-Shot 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	ViLBERT	~900	-	-	-	-	-	-	31.9	61.1	72.8	-	-	-
	Unicoder-VL	~925	64.3	85.8	92.3	-	-	-	48.4	76.0	85.2	-	-	-
	UNITER-Base	~900	80.7	95.7	98.0	-	-	-	66.2	88.4	92.9	-	-	-
	ImageBERT†	~925	70.7	90.2	94.0	44.0	71.2	80.4	54.3	79.6	87.5	32.3	59.0	70.2
Linear	ViLT-B/32	~15	69.7	91.0	96.0	53.4	80.7	88.8	51.3	79.9	87.9	37.3	67.4	79.0
	ViLT-B/32⊕	~15	73.2	93.6	96.5	56.5	82.6	89.6	55.0	82.5	89.8	40.4	70.0	81.1

Fine-tuning

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

• Experiments

❖ Ablation Study : Whole Word Masking, Masked Patch Prediction, RandAugment

- Whole Word Masking : 한 단어 전체를 Masking, 저자들이 사용한 토크나이저는 원래 전체 마스킹이 안됨.

  ex. 'giraffe' masking : **Before** ['gi', '##raf', '##fe'] → ['gi', '[mask]', '##fe'], **After** ['gi', '##raf', '##fe'] → [mask]

- Masked Patch Prediction(MPP) : Patch에 Masking을 씌워 출력 단계에서 Masking된 Patch의 RGB값을 예측하는 Task

  ablation study 결과, Masked Patch Prediction은 학습에 큰 영향을 주지 못하는 것으로 확인됨(오히려 성능이 저하)

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	NLVR2 test-P	Flickr30k R@1 (1K) TR (ZS)	Flickr30k R@1 (1K) IR (ZS)	MSCOCO R@1 (5K) TR (ZS)	MSCOCO R@1 (5K) 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)

• Experiments

❖ Visualization

- 단어와 Patch간의 Attention Score를 시각화

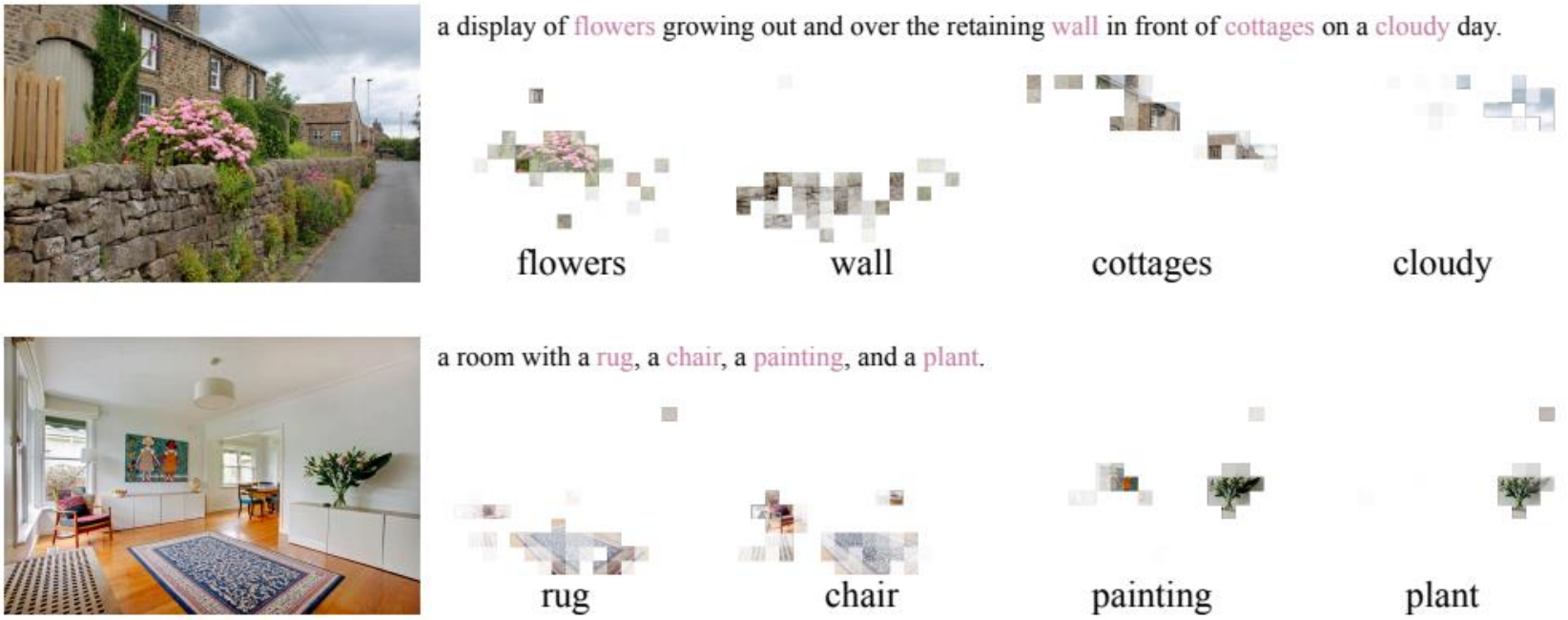


Figure 4. Visualizations of transportation plan of word patch alignment. Best viewed zoomed in.

• Conclusion

❖ PSViT : Better Vision Transformer via Token Pooling and Attention Sharing

- 트랜스포머가 중복된 특징을 뽑아내는 것을 발견하고 Token pooling과 Attention sharing이라는 간단한 기법을 적용하여 성능을 향상시킴
- 최적의 모델 구조를 찾기 위해서 AutoML을 적용

- 후기
  - ➢ Attention sharing의 효과를 입증했고 연산량을 줄여주지만 feature의 redundancy를 줄이는 것인지는 의문(같은 attention score를 공유하고 비슷한 embedding feature면 redundant한 특징이 추출되는 것이 아닌가?)
  - ➢ ViT가 점점 CNN과 닮아지고 있다는 생각이 들었음
  - ➢ Weight-based AutoML이 어떤 알고리즘으로 진행되는지 궁금

• Appendix

Wasserstein distance

- **Earth mover's distance (Wasserstein-1 distance)**

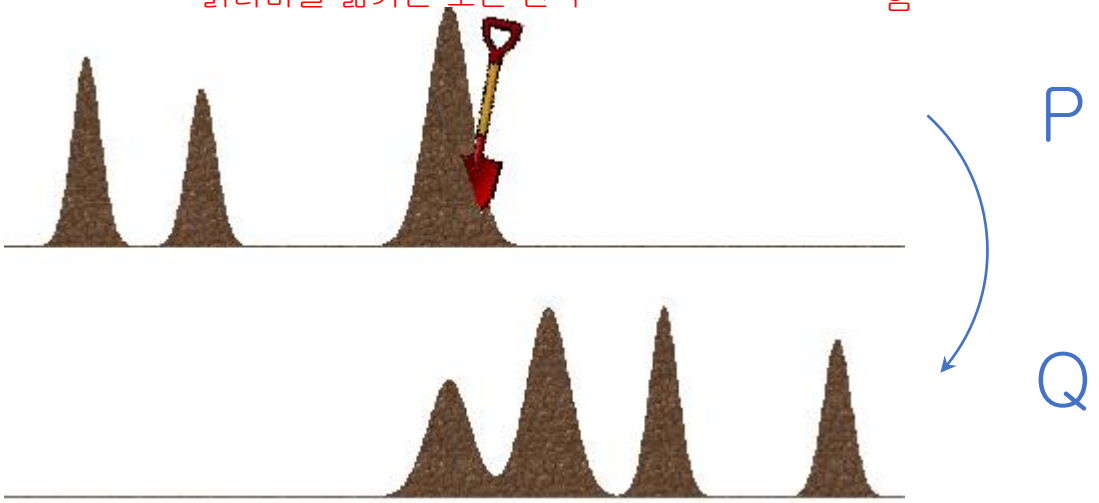
모든 전략 중 최소한의 힘

$$W(P,Q) = \inf_{\gamma \epsilon (P,Q)} E_{(x,y)\sim\gamma}[\|x-y\|]$$

최소화

흙더미를 옮기는 모든 전략

흙더미를 옮길 때 필요한 힘



- **여기서 $\prod(\mathbb{P},\mathbb{Q})$는 두 확률분포 $\mathbb{P},\mathbb{Q}$ 의 결합확률분포(joint distribution)들을 모은 집합이고 γ는 그 중 하나의 원소. 즉 모든 결합확률분포 $\prod(\mathbb{P},\mathbb{Q})$ 중에서 d(X,Y) 의 기대값을 가장 작게 추정한 값을 의미.**

• Reference

ViLT : Vision-and-Language Transformer Without Convolution or Region Supervision

1. Kim, W., Son, B., & Kim, I. (2021). Vilt: Vision-and-language transformer without convolution or region supervision. *arXiv preprint arXiv:2102.03334*.
2. Appendix 관련 참조 : https://www.slideshare.net/ssuser7e10e4/wasserstein-gan-i

KOREA UNIVERSITY

DMQA