😀

ViLBERT : Pretraining Task-Agnostic Visiolinguistic Representations

▼ BERT

BERT의 경우 transformer의 encoder만을 사용한 모델이다. Transfromer의 encoder의 문장 이해력을 활용하여 QA, 문장 관계 판단등 "이해"를 기반으로 한 테스크에 적합한 모델이다.

Abstract

```
 이미지 ----------------------------> Faster R-CNN --------------> [Object Fe
ex) 이미지 → Faster R-CNN → [obj1: 고양이 얼굴], [obj2: 창문], [obj3: 손]
                                                ↘
                                                 Co-Attention Layer → Joint Embedding
                                                ↗
텍스트 [문장] → Tokenizer → BERT-style Text Stream → [Text Features]
```

ViLBERT는 이미지와 자연어 간의 공동 표현(joint representation)을 학습하기 위한 멀티모달 모델로, 특정 작업(task)에 종속되지 않는 일반적인 표현 학습을 목표로 합니다. 이를 위해 이미지와 텍스트는 각각 독립적인 스트림(two-stream)으로 처리되며, 텍스트 스트림은 기존 BERT 구조를 거의 그대로 따르고, 이미지 스트림은 **Faster R-CNN**과 같은 객체 탐지 모델을 활용하여 **의미 있는 객체 수준의 시각적 특징**을 추출합니다.

이렇게 추출된 이미지 객체 벡터들과 텍스트 토큰 벡터들은 **co-attention 메커니즘**을 통해 서로 간의 정보를 교환하며, 두 모달리티 간의 연관성을 학습합니다. 이 과정을 통해 ViLBERT는 이미지와 텍스트를 공통된 의미 공간(shared vector space)으로 매핑할 수 있는 표현을 생성하게 됩니다.

🍎 즉, ViLBERT는 단순히 하나의 태스크(예: VQA 등)를 위한 멀티모달 연결이 아닌, **이미지와 언어 간의 일반적인 의미 연결(visual grounding)을 사전학습(pretraining) 가능한**

능력으로 확장하고 있다는 점에서 큰 의의가 있습니다. 이러한 기반 위에서, 다양한 태스크에 파인튜닝(fine-tuning)을 적용함으로써 **우수한 성능을 달성**할 수 있음을 보여주었습니다.

Introduction

이전의 연구들은 모두 잘 사전학습된 vision model과 language model을 그대로 가져와 수행하고 싶은 Task에 최적화되도록 fine tuning 시키면서 발전되었습니다. 하지만 본 논문에서는 이렇게 학습된 모델이 😈실제로 언어와 이미지간의 관계를 파악한것인가? 를 문제로 제기하였습니다. 그리고 이러한 방법들은 만일 충분한 데이터가 존재하지 않는다면 일반화가 어렵다고 주장하고 있습니다.

이에 해당 논문에서는 위의 문제를 개선하기 위해서 이미지와 자연어의 관계 자체를 학습할 수 있는 모델을 제안합니다. BERT가 NLP 에서 문맥 자체를 일반화 한것 처럼 이미지와 자연어의 관계도 사전학습 가능하다고 제안하고 있습니다.

> *이미지-자연어 관계 사전학습?*
> 이미지와 관련된 문장을 보고 어떤 단어가 어떤 객체/행동과 연결되는지를 미리 학습할 수 있는 모델을 의미합니다. 즉, 특정 Task 없이도, 자전거라는 단어가 들어왔을때, 이미지 내에서 자전거와 관련된 객체를 주목하도록 학습 후, 이를 다양한 Task에 적용할 수 있는 모델을 의미합니다.

해당 논문에서는 이미지-자연어의 간의 joint-visual-linguistic representation을 학습하기 위해서 self-supervised learning의 성공 사례를 참고합니다. 이를 위해서 “proxy task”를 도입하게됩니다. “proxy task”란 라벨이 존재하지 않는 경우 내부의 패턴이나 구조를 활용하여 하위 테스크를 만들어 자동으로 학습하는 방법을 의미합니다. 예를 들어 BERT에서는 문맥 자체를 이해하기 위해서 특정 단어를 [MASK] 하여 해당 단어를 예측하는 proxy task를 사용한 좋은 사례입니다. ViLBERT 또한 이와 유사한 방식으로 이미지와 자연어 간의 의미 연결(visual grounding)을 학습하고자 하며, 이를 위해서 330만개의 이미지와 캡션의 쌍으로 구성된 Conceptual Caption 데이터셋 을 사용하였다고 합니다.

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.

Alt-text: Musician Justin Timberlake performs at the 2017 Pilgrimage Music & Cultural Festival on September 23, 2017 in Franklin, Tennessee.

Conceptual Captions: pop artist performs at the festival in a city.

Figure 1: Examples of images and image descriptions from the Conceptual Captions dataset; we start from existing alt-text descriptions, and automatically process them into Conceptual Captions with a balance of cleanliness, informativeness, fluency, and learnability.

Conceptual caption dataset example

본 논문에서는 Vision & Language BERT (ViLBERT) 모델을 제안합니다. 기존의 BERT를 확장하여 visual grounding 학습이 가능하도록 설계되었습니다. 해당 모델의 핵심은 이미지와 자연어가 각각의 독립된 stream을 통과하여 co-attention layer에서 상호작용할 수 있도록 설계 되었다는 것입니다. 이렇게 two-stream으로 설계함으로 써 이후 각 모달리티에 최적화된 방법을 적용하기에 유리하며, 단일 stream을 사용한 모델보다 더 나은 성능을 보여줌을 제안합니다. 그리고 2가지 proxy task로 학습하였다고 합니다.

(1) **이미지와 자연어 입력에서 마스킹된 요소를 예측하는 과제** (Masked Region & Token Prediction)

(2) **주어진 이미지와 문장이 서로 연관된 쌍인지 판단하는 이진 분류 과제** (Image-Text Matching)

이렇게 사전학습된 ViLBERT는 VQA, VCR, Referring Expression, Caption-based Image Retrieval 등 여러 대표적인 vision-and-language 하위 과제에 적용되며, 모든 태스크에서 state-of-the-art(SOTA) 성능을 달성합니다. 기존의 task-specific 모델들과 비

교했을 때, 2%에서 최대 10%까지 성능이 향상되었으며, ViLBERT 구조는 다양한 과제에 간단한 수정만으로도 적용 가능한 범용 기반 모델로 활용됩니다.

Introduction 정리

BERT의 성공에 힘입어, 본 논문에서는 **이미지-자연어 관계 자체를 사전학습할 수 있는 ViLBERT**를 제안합니다. ViLBERT는 **2-stream 구조**를 기반으로 하여, 각 모달리티에 특화된 모델을 사용할 수 있다는 장점을 가지며, 이후 **co-attention 메커니즘을 통해 시각과 언어 정보가 상호작용**하도록 구성됩니다. 학습은 **두 가지 proxy task**를 통해 수행되며, 사전학습된 모델은 **fine-tuning을 통해 다양한 vision-and-language task에서 높은 성능을 보여줍니다.**

2줄 정리

ViLBERT는 BERT 구조를 확장하여 이미지와 자연어 간 관계를 사전학습하며, 2-stream + co-attention 구조로 다양한 Task에 효과적으로 전이 가능한 멀티모달 모델입니다.

2. Approach

2.1 Preliminaries : Bidirectional Encoder Representation form Transformers (BERT)

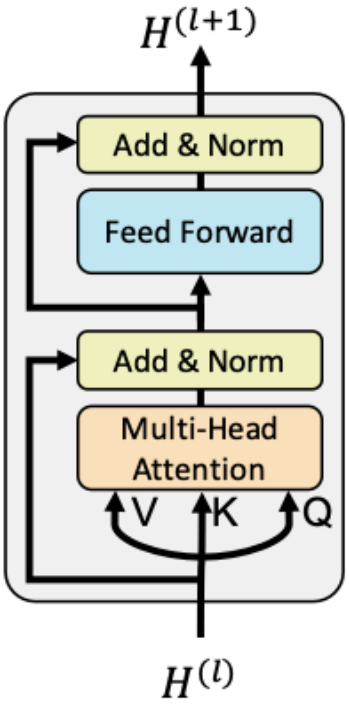
ViLBERT를 이해하기에 앞서, 논문에서는 먼저 BERT에 대해 간단히 언급하고 있습니다.

BERT는*양방향 언어 모델(bidirectional language model)로, 대규모 텍스트 코퍼스를 기반으로 사전학습(pretraining)되어 문맥(Context)을 효과적으로 학습합니다.

이후에는 각 태스크에 맞게 **미세조정(fine-tuning)** 되어 활용됩니다. BERT의 입력은 다음 **세 가지 임베딩의 합**으로 구성됩니다:

- **Token Embedding**: 단어 자체의 의미 벡터

- **Positional Embedding**: 시퀀스 내에서의 위치 정보
- **Segment Embedding**: 문장이 여러 개인 경우, 문장 구분을 위한 정보



(a) Standard encoder transformer block

이 임베딩 합산 결과는 **다층 Transformer Encoder**로 입력되며, 각 층은 **Multi-Head Attention**과 **Feedforward Network**로 구성되어 있습니다.

BERT는 다음의 **두 가지 proxy task**를 통해 사전학습됩니다:

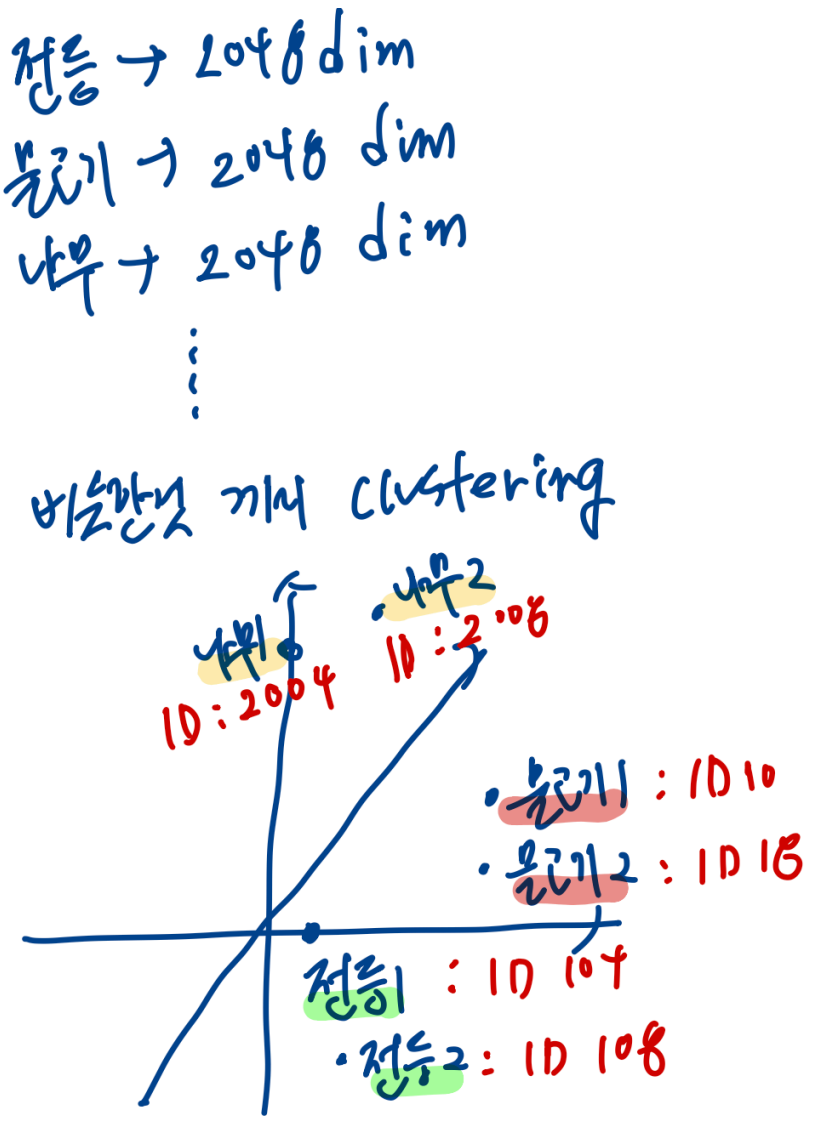
1. **Masked Language Modeling (MLM)**: 입력 문장에서 일부 단어를 [MASK] 처리한 후, 해당 단어를 예측
2. **Next Sentence Prediction (NSP)**: 두 문장 A, B가 주어졌을 때, B가 실제로 A 다음에 나오는 문장인지(True/False) 예측

이 두 과제는 각각 **Cross-Entropy Loss** (MLM)와 **Binary Cross-Entropy Loss** (NSP)를 통해 학습됩니다.

2.2 ViLBERT : Extending BERT to Jointly Represent Images and Text

우선 가장 쉬운 접근 방법에 대해 소개합니다. 가장 쉬운 접근 방법은 이미지 내 객체를 판별하고 이를 clustring하여 ID를 매핑하게 되면 이산화된 vector로 이미지를 표현할 수 있고 이를 단순히 BERT의 입력에 연결하는 아이디어 입니다. 이를 그림으로 표현하면 간다하게 아래와 같이 표현해 볼 수 있습니다.

다음과 같이 이미지에서 객체가 있을법한 위치를 추출하고, 각 이미지를 d차원으로 임베딩 합니다. 그리고 각 clustering을 활용하여 비슷한 위치에 있는 객체에 비슷한 ID를 제공하게 됩니다.



이를 통해서 이미지가 최종적으로 [2004,2008,10,104,108,18] 이런식으로 이산화된 벡터로 표현할 수 있게 되고 이를 단순히 [image_t1, image_t2, ..., text_t1, text_t2.....] 를 BER의 입력으로 사용할 수 있게 됩니다.

하지만 이러한 방식을 사용하게 되면 아래와 같은 문제점이 발생합니다.

(1) 단어와 이미지의 임베딩 차원이 다르지만 이를 무시하게된다.

(2) 이미지를 단순히 ID로 임베딩하게 되면 공간정보를 잃게 된다.

(3) 모달리티 특징 학습이 어려워진다.

(4) 기존 BERT를 활용할 수 없고 성능 저하로 이어질 수 있다.

이를 방지하기 위해서 해당 논문에서는 2-stream을 제안합니다.

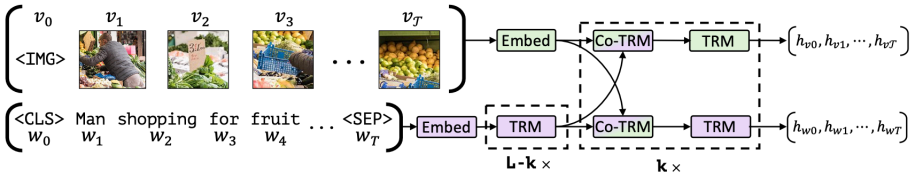


Figure 1: Our ViLBERT model consists of two parallel streams for visual (green) and linguistic (purple) processing that interact through novel co-attentional transformer layers. This structure allows for variable depths for each modality and enables sparse interaction through co-attention. Dashed boxes with multiplier subscripts denote repeated blocks of layers.

이미지 :

이미지의 경우 Faster R-CNn과 같이 각 객체를 탐지할 수 있는 모델을 통해서 각 객체가 d 차원으로 임베딩 됩니다. 그리고 각각 벡터는 V_i로 표현됩니다.

자연어 :

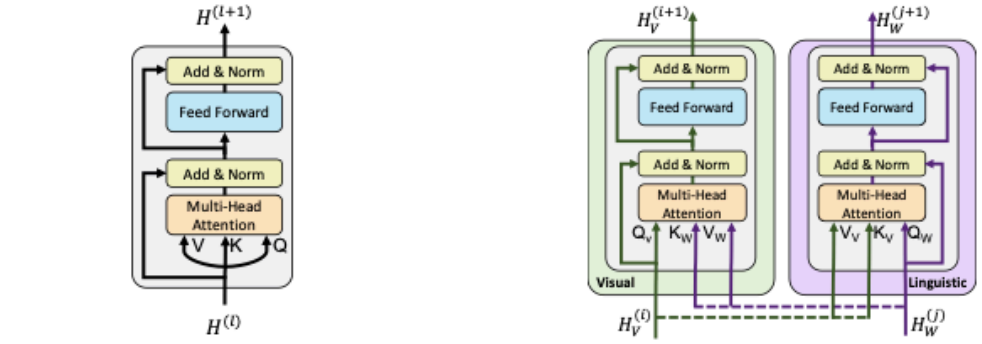
자연어의 경우 BERT와 동일한 방식으로 토큰화 및 임베딩되며, Transformer(TRM) 층을 거치면서 단어 간 문맥 정보를 학습합니다. 이미지 특징이 이미 고차원 추상 표현이기 때문에, 텍스트 역시 이를 대응할 수 있도록 충분한 표현력을 가진 벡터로 변환되도록 TRM을 여러 층 거칩니다.

Co-Attention 및 후처리 :

이후 embedding된 모달리는 co-Attention을 통해서 상호작용하게 되고, 이후 각각 TRM 층을 통과하며 다시한번 문맥을 정제하게 됩니다. 이를 통해서 이미지는 문맥을 파악안 객체들의 벡터, 자연어는 이미지를 파악한 단어들의 벡터를 얻을 수 있게 됩니다.

추가로 본 논문에서 이미 두 모달리티가 충분히 특징을 잘 포착하였기 때문에 특정 층에서만 상호작용을 제한적으로 수행해도 된다는 직관을 반영하였다고 합니다. (즉, 지속적으로 이미지와 단어를 상호작용 시킬 필요가 없다는 것을 의미합니다.)

Co-Attention Transformer Layers



(a) Standard encoder transformer block (b) Our co-attention transformer layer

Figure 2: We introduce a novel co-attention mechanism based on the transformer architecture. By exchanging key-value pairs in multi-headed attention, this structure enables vision-attended language features to be incorporated into visual representations (and vice versa).

다음으로 co-attention layer에 대해 설명하고 있습니다. co-attention 또한 기존의 attention과 마찬가지로 Q,K,V를 통해서 행렬 계산이 진행됩니다. 하지만 Q와 (K,V)가 서로 다른 모달리티에서 온다는 것이 중요한 포인트입니다. Image stream에서는 이미지를 기반으로 언어들과 attention을 진행하게 되고, Language stream에서는 언어 문맥을 기반으로 이미지와 attention을 진행하게 됩니다. 나머지 부분들은 기존의 transformer의 작동 방식과 다르지 않습니다. 이러한 co-attention을 통해서 각 스트림은 상대 모달리티의 정보를 반영한 Joint representation을 생성하게 됩니다.

> Image Stream : Q (이미지 벡터) , K,V(언어 벡터)
> Language Stream : Q (언어 벡터) , K,V(이미지 벡터)

Image Representation

이미지 데이터의 경우 사전 학습된 객체 탐지 모델에서 나온 각 bounding box들을 사용하였습니다. 각각의 객체들은 아래와 같이 총 5D로 변환됩니다.

- 좌상단 (x_1, y_1)
- 우하단 (x_2, y_2)
- 이미지 내 면적 비율 (area ratio)

그리고 해당 벡터는 이미지 feature map과 동일 차원으로 선형변환된 후 feature map과 덧셈을 통해서 최종 image vector를 생성하게 됩니다. 그리고 [IMG] 토큰을 추가하여 이미지 전체에 대한 정보 또한 추가해 줍니다. 그래서 입력의 경우 아래와 같이 나타나게 됩니다.

{[IMG], v_1, ..., v_T, [CLS], w_1, ..., w_T, [SEP]}
IMG : 전체 이미지에 대한 정보를 담고 있다.
CLS : 전체 문장에 대한 정보를 담고 있다.

Training Tasks and OBjecteives

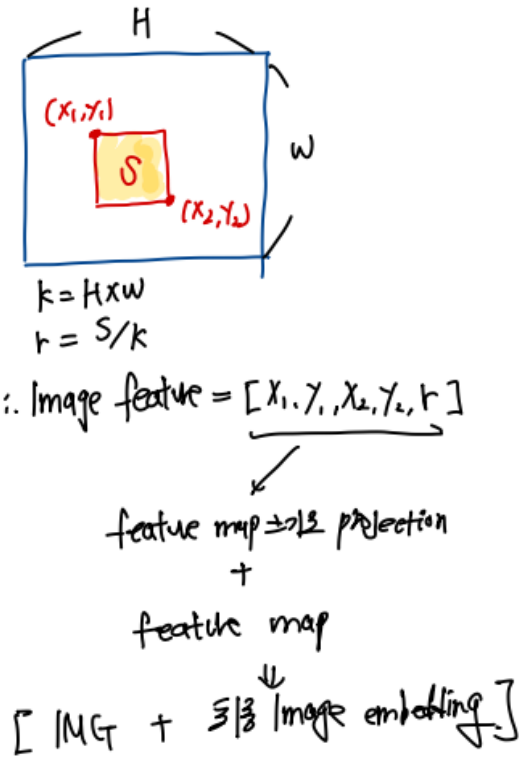
총 2개의 proxy task를 통해서 Joint visual-linguistic representation을 학습합니다.

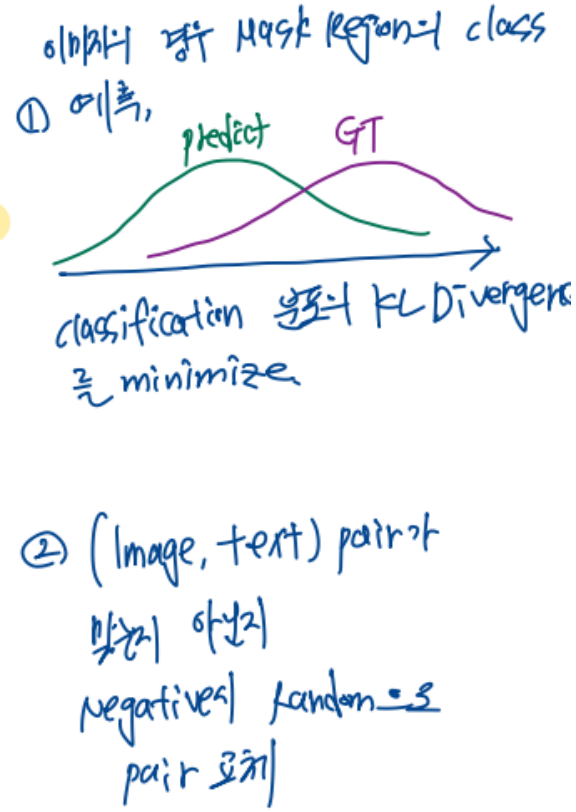
(1) [MASK]를 하여 해당 위치 예측

자연어의 경우 단순히 예측하지만 이미지의 경우 마스킹 처리된 영역의 클래스를 맞추는 clasification을 수행하게 됩니다. 그리고 예측 분포를 활용하여 정답과 KL DIvergence를 최소화하는 방향으로 학습되게 됩니다.

(2) [이미지, 단어] 쌍 맞추기

만일 Negative인 경우 랜덤하게 바꿔가면서 학습하게 됩니다. 간단하게 두 벡터의 선형 곱을 한후 간단한 네트워크를 통과하여 1/0을 예측하게 됩니다.





정리

ViBERT는 2-stream구조와 co-attentnion 매커니즘을 통해서 이미지-자연어 사이의 joint-representation을 학습하며 MMA와 MMM 2개의 proxy-task를 통해서 사전학습을 합니다.
이후 각 Task에 맞는 Fintunning을 통해 각 Task에서 높은 성능을 보여줍니다.

Result and Analysis

[기존 SOTA 모델들과의 비교]

기존의 모델들보다 더 높은 성능을 보여줌.

		VQA [3]	VCR [25]			RefCOCO+ [32]			Image Retrieval [26]			ZS Image Retrieval		
	Method	test-dev (test-std)	Q→A	QA→R	Q→AR	val	testA	testB	R1	R5	R10	R1	R5	R10
SOTA	DFAF [36]	70.22 (70.34)	-	-	-	-	-	-	-	-	-	-	-	-
SOTA	R2C [25]	-	63.8 (65.1)	67.2 (67.3)	43.1 (44.0)	-	-	-	-	-	-	-	-	-
SOTA	MAttNet [33]	-	-	-	-	65.33	71.62	56.02	-	-	-	-	-	-
SOTA	SCAN [35]	-	-	-	-	-	-	-	48.60	77.70	85.20	-	-	-
Ours	Single-Stream†	65.90	68.15	68.89	47.27	65.64	72.02	56.04	-	-	-	-	-	-
Ours	Single-Stream	68.85	71.09	73.93	52.73	69.21	75.32	61.02	-	-	-	-	-	-
Ours	ViLBERT†	68.93	69.26	71.01	49.48	68.61	75.97	58.44	45.50	76.78	85.02	0.00	0.00	0.00
Ours	ViLBERT	**70.55 (70.92)**	**72.42 (73.3)**	**74.47 (74.6)**	**54.04 (54.8)**	**72.34**	**78.52**	**62.61**	**58.20**	**84.90**	**91.52**	**31.86**	**61.12**	**72.80**

[Task 별 TRM 계층 비교]

각 테스크 마다 최적의 TRM 갯수가 다르다 → 각 테크스 마다 필요한 문맥 통합 수준이 다르다는 것을 암시.

	VQA [3]	VCR [25]			RefCOCO+ [32]			Image Retrieval [26]			ZS Image Retrieval [26]		
Method	test-dev	Q→A	QA→R	Q→AR	val	testA	testB	R1	R5	R10	R1	R5	R10
ViLBERT (2-layer)	69.92	72.44	**74.80**	**54.40**	71.74	**78.61**	62.28	55.68	84.26	90.56	26.14	56.04	68.80
ViLBERT (4-layer)	70.22	**72.45**	74.00	53.82	72.07	78.53	**63.14**	55.38	84.10	90.62	26.28	54.34	66.08
ViLBERT (6-layer)	**70.55**	72.42	74.47	54.04	**72.34**	78.52	62.61	58.20	84.90	**91.52**	31.86	61.12	72.80
ViLBERT (8-layer)	70.47	72.33	74.15	53.79	71.66	78.29	62.43	**58.78**	**85.60**	91.42	**32.80**	**63.38**	**74.62**

[학습 데이터 양에 따른 성능 비교]

데이터를 많이 사용하면 할수록 더 높은 성능을 보임 → 이에 대량 모델로 확장 가능성을 제시함.

Table 3: Transfer task results for ViLBERT as a function of the percentage of the Conceptual Captions dataset used during pre-training. We see monotonic gains as the pretraining dataset size grows.

	VQA [3]	VCR [25]			RefCOCO+ [32]			Image Retrieval [26]			ZS Image Retrieval [26]		
Method	test-dev	Q→A	QA→R	Q→AR	val	testA	testB	R1	R5	R10	R1	R5	R10
ViLBERT (0 %)	68.93	69.26	71.01	49.48	68.61	75.97	58.44	45.50	76.78	85.02	0.00	0.00	0.00
ViLBERT (25 %)	69.82	71.61	73.00	52.66	69.90	76.83	60.99	53.08	80.80	88.52	20.40	48.54	62.06
ViLBERT (50 %)	70.30	71.88	73.60	53.03	71.16	77.35	61.57	54.84	83.62	90.10	26.76	56.26	68.80
ViLBERT (100 %)	**70.55**	**72.42**	**74.47**	**54.04**	**72.34**	**78.52**	**62.61**	**58.20**	**84.90**	**91.52**	**31.86**	**61.12**	**72.80**

또한 언어와 이미지 간 의미적 매핑이 이뤄졌는지 확인하기 위해서 zero-shot실험을 진행하였고

- 사전학습만 된 ViLBERT: R1 = 31.86
- 기존 SOTA (Flickr30k 학습 O): R1 = 48.60
- ViLBERT fine-tuning O: R1 = 58.20

다음과 같은 결과를 얻었다고 합니다. 이를 통해서 fine-tunning 없이도 꽤 높은 성능을 보여주며 이미지와 문장 사이에 유의미한 관계를 맺고 있음을 보여주고 있습니다. 이를 통해서 ViBERT 모델이 추후 zero-shot learning 으로의 확장 가능성을 보여주고 있습니다.

Conclusion

본 논문에서는 이미지와 자연어 간의 Joint Representation을 학습하는 모델을 제안하고, 대규모 자동 수집된 데이터를 활용하여 사전 학습을 통해서 Visual grounding 능력을 학습하였습니다.

간단하게 2-stream 구조와 co-attention 아키텍처를 도입하여 다양한 Task 에서 Fine Tunning 을 통해서 SOTA를 달성함을 통해 앞으로 멀티 테크스 학습에 확장가능성을 보여줍니다.

내 생각

Vision Transformer와 다르게 이미지의 영역을 단순히 나눠서 토큰화 시키는 것이 아니라, 객체 탐지 모델을 활용하여 각각의 객체들을 토큰화 하는 아이디어가 정말 좋은 것 같다. 그리고 서로 다른 모달의 특징 차원까지 고려하여 독립적으로 학습하게 하는 것 또한 좋은 아이디어인 것 같다.

여기서 조금 더 제안해볼만한 사항들

1. 객체를 최대 12 ~ 36개로 설정한다. 하지만 만일 크게 디테일하지 않은 차원에서 고양이가 50마리 있는 경우 고양이 이미지에 대해서만 추출하게된다. 그래서 Instance segmentation 처럼 클래스의 값이 동일하다면 가장 높은 N개만 뽑아서 추출하도록 하는 방법을 사용하게 되면 객체가 많은 이미지에 대해서 더 높은 성능을 보여줄 수 있을것 같다.
2. 본 논문에서는 둘다 Transformer의 encoder만 사용하는 구조를 가지고 있는데, 순서의 중요도가 떨어지는 이미지의 경우 encoder를 사용하고 순서가 중요한 자연어의 경우 deocder를 사용하여 스토리 텔링 처럼 이미지의 context를 보고 문장을 완성하는 모델 또한 좋은 아이디어가 될 수 있지 않을까? 생각이 든다. proxy task는 이미지는 본 논문과 동일하고, 자연어의 경우 다음 단어예측을 진행하면 충분히 가능하다고 생각이 든다.
3. [CLS]와 [IMG] 토큰이 모델 성능에 크게 도움이 되는지 의문이다. 이미 각각 객체에 대한 정보가 담겨져있는데, 굳이 이미지와 문장 전체에 대한 벡터를 사용하는 이유에 대해서 의문이든다.