GIT: A Generative Image-to-text Transformer for Vision and Language

https://arxiv.org/abs/2205.14100

0. Introduction

- 이미지와 텍스트 간 멀티모달 생성 모델 연구에 집중함
- 기존 이미지-텍스트 모델들은 주로 인식이나 매칭에 초점 맞춤
- 본 논문은 이미지에서 텍스트를 생성하는 생성적(transformer 기반) 모델인 GIT 제안함
- GIT은 다양한 비전-언어 태스크에서 높은 생성 품질과 범용성 보여줌
- 이미지의 시각 정보를 효과적으로 텍스트로 변환하는 새로운 구조와 학습 방법 도입함
- 멀티태스크 학습과 대규모 데이터셋 활용으로 성능 극대화함

1. Overview

- GIT는 이미지 입력을 받아 텍스트를 생성하는 transformer 기반 모델임
- 이미지 인코더와 텍스트 디코더로 구성됨
- 인코더는 이미지 특징을 추출하고, 디코더는 이를 바탕으로 문장 생성함
- 멀티태스크 학습 프레임워크로 이미지 캡셔닝, 비주얼 QA 등 다양한 태스크 처리 가능함
- 대규모 비전-언어 데이터셋으로 사전학습 수행함
- 생성 품질과 태스크 적응력 모두 뛰어남을 보임

2. Challenges

- 이미지와 텍스트 간 복잡한 의미 관계 학습이 어려움

- 이미지에서 자연스러운 문장 생성이 까다로움
- 다양한 태스크에 유연하게 대응하는 모델 설계가 필요함
- 대규모 데이터와 계산 자원이 요구돼 학습 비용이 높음
- 멀티태스크 학습 시 태스크 간 간섭 문제 존재함
- 생성 모델이 과적합이나 반복 생성 문제에 취약할 수 있음

3. Method

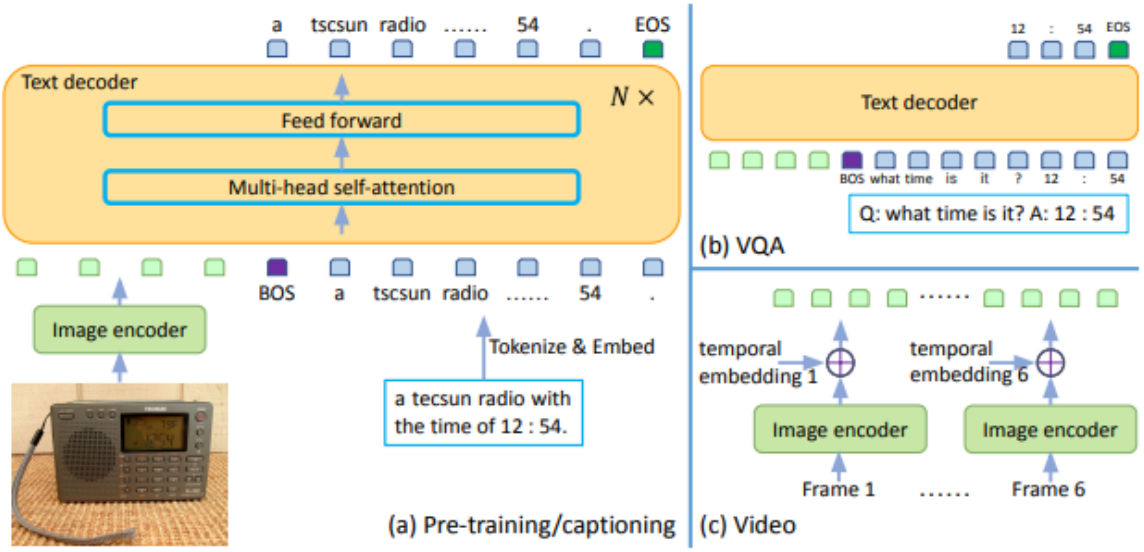


Figure 2: Network architecture of our GIT, composed of one image encoder and one text decoder. (a): The training task in both pre-training and captioning is the language modeling task to predict the associated description. (b): In VQA, the question is placed as the text prefix. (c): For video, multiple frames are sampled and encoded independently. The features are added with an extra learnable temporal embedding (initialized as 0) before concatenation.

- 이미지 인코더로 CNN 또는 비전 트랜스포머 사용함
- 텍스트 디코더는 autoregressive transformer 구조임
- 이미지 특징을 텍스트 생성에 효과적으로 연결하는 cross-modal attention 적용함
- 멀티태스크 학습 프레임워크로 다양한 비전-언어 태스크 통합 학습함
- 대규모 데이터셋으로 사전학습 후 태스크별 미세조정 수행함
- 학습 안정성과 성능을 높이기 위해 여러 정규화와 데이터 증강 기법 활용함
- 생성 품질 향상을 위해 교사 강제 학습(teacher forcing)과 기타 기법 적용함

4. Experiments

- MSCOCO, Flickr30k, VQA 등 대표 비전-언어 데이터셋 사용함
- 이미지 캡셔닝, 비주얼 QA, 이미지-텍스트 매칭 등 태스크 평가함
- 기존 SOTA 모델들과 성능 비교함
- 다양한 크기 모델과 학습 설정에 따른 ablation study 진행함
- 멀티태스크 학습 효과와 데이터셋 크기 영향 분석함
- 생성 텍스트 품질을 자동 및 인간 평가 방식으로 검증함
- 학습 속도와 자원 효율성도 함께 측정함

5. Results

	Image captioning				Image QA			Video captioning				Video QA		Text Rec.
	COCO*	nocaps*	VizWiz*	TextCaps*	ST-VQA*	VizWiz*	OCR-VQA	MSVD	MSRVTT	VATEX*	TVC*	MSVD-QA	TGIF-Frame	Avg on 6
Prior SOTA[1]	138.7	120.6	94.1	109.7	69.6	65.4	67.9	120.6	60	86.5	64.5	48.3	69.5	93.8
GIT (ours)	148.8	123.4	114.4	138.2	69.6	67.5	68.1	180.2	73.9	93.8	61.2	56.8	72.8	92.9
Δ	+10.1	+2.8	+20.3	+28.5	+0.0	+2.1	+0.2	+59.6	+13.9	+7.3	-3.3	+8.5	+3.3	-0.9
GIT2 (ours)	149.8	124.8	120.8	145.0	75.8	70.1	70.3	185.4	75.9	96.6	65.0	58.2	74.9	94.5
Δ	+11.1	+ 4.2	+26.7	+35.3	+6.2	+4.7	+2.4	+64.8	+15.9	+10.1	+0.5	+9.9	+5.4	+0.7

Table 3: Zero/Few/Full-shot evaluation on Flickr30K with Karpathy split.

Shot	0	16	32	290 (1%)	full
Zhou et al. (2020)	-	-	-	-	68.5
Flamingo	67.2	78.9	75.4	-	-
GIT	49.6	78.0	80.5	86.6	98.5

Vocabulary	Method	test-std
Closed	OSCAR	73.82
	UNITER	74.02
	VILLA	74.87
	UNIMO	75.27
	ALBEF	76.04
	VinVL	76.60
	UFO	76.76
	CLIP-ViL	76.70
	METER	77.64
	BLIP	78.32
	SimVLM (-, 1.8B)	80.34
	Florence (0.9B, 14M)	80.36
	mPlug (0.6B, 14M)	81.26
	OFA (0.9B, 54M)	82.0
	CoCa (2.1B, 4.8B)	**82.3**
Open	Flamingo (80B, 2.3B)	**82.1**
	GIT (0.7B, 0.8B)	78.81

(a) VQAv2

Method	test
M4C	40.46
LaAP-Net	41.41
SA-M4C	44.6
SMA	45.51
TAP	53.97
Flamingo	54.1
Mia	**73.67**
GIT	59.75

(b) TextVQA

Method	test
(Liu et al., 2021)##	60.6
Flamingo	65.4
GIT	**67.5**

(c) VizWiz-QA

Method	Test ANLS
M4C	46.2
SMA	46.6
CRN	48.3
LaAP-Net	48.5
SA-M4C	50.4
TAP	59.7
LaTr	**69.6**
GIT	**69.6**

(d) ST-VQA

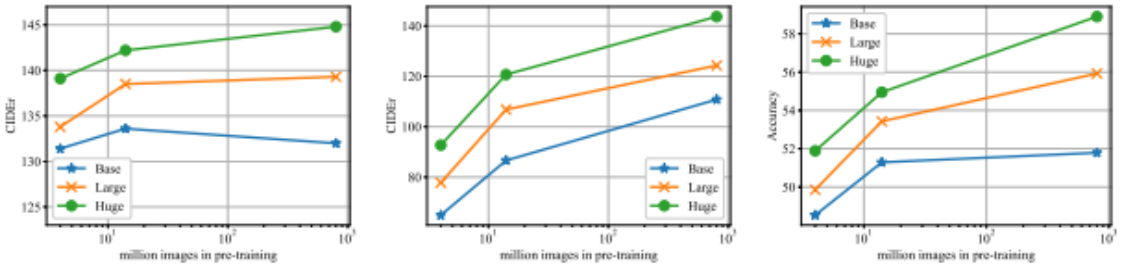
Method	test
BLOCK+CNN+W2V	48.3
M4C	63.9
LaAP-Net	64.1
LaTr	67.9
GIT	**68.1**

(e) OCR-VQA

Vocabulary	Method	Top-1
Closed	ALIGN (Jia et al., 2021)	88.64
	Florence (Yuan et al., 2021)	90.05
	CoCa (Yu et al., 2022)	**91.0**
Open	GIT	88.79

Method	FT data	Average
SAM (Liao et al., 2019)	MJ+ST	87.8
Ro.Scanner (Yue et al., 2020)	MJ+ST	87.5
SRN (Yu et al., 2020)	MJ+ST	89.6
ABINet (Fang et al., 2021a)	MJ+ST	91.9
S-GTR (He et al., 2022b)	MJ+ST	91.9
MaskOCR (Lyu et al., 2022)	MJ+ST	**93.8**
GIT	TextCaps	89.9
	MJ+ST	92.9

	Zero-shot			1-shot per class			5-shot per class		
Accuracy type	equal	in	voc-prior	equal	in	voc-prior	equal	in	voc-prior
Flamingo	-	-	-	-	-	71.7	-	-	77.3
GIT	1.93	40.88	33.48	64.54	66.76	72.45	79.79	80.15	80.95



(a) COCO (b) TextCaps (c) VizWiz-QA

Figure 4: Performance with different pre-training data scales and different model sizes.

Layers	COCO				nocaps	
	B@4	M	C	S	C	S
6	38.9	30.7	136.4	24.6	119.3	15.9
12	38.9	30.6	136.0	24.2	118.1	15.5
24	39.1	30.2	134.6	23.8	115.4	15.1

- GIT 모델이 이미지 캡셔닝, VQA 등 여러 태스크에서 기존 SOTA 성능 능가함
- 멀티태스크 학습으로 태스크 간 시너지 효과 나타남
- 대규모 데이터 사전학습이 성능 향상에 크게 기여함
- ablation study에서 인코더, 디코더 구조와 학습 전략의 중요성 확인됨
- 자동 평가 지표와 인간 평가 모두에서 생성 텍스트 품질 우수함
- 학습 효율성도 기존 모델 대비 개선됨

6. Insight

- 이미지에서 텍스트를 생성하는 멀티모달 생성 모델로서 transformer 구조가 효과적임
- 멀티태스크 학습이 다양한 비전-언어 태스크에 유연하게 대응 가능하게 함
- 대규모 데이터와 사전학습이 생성 품질과 성능을 크게 끌어올림
- cross-modal attention과 학습 기법들이 모델 성능과 안정성에 기여함
- 하지만 학습 비용과 계산 자원 요구가 높아 실무 적용에 부담일 수 있음
- 태스크 간 간섭 문제와 과적합 위험도 존재함
- 향후 경량화와 효율적 학습 방법 연구가 필요함