

NVLM: Open Frontier–Class Multimodal LLMs

Wenliang Dai* Nayeon Lee* Boxin Wang* Zhuolin Yang*
Zihan Liu Jon Barker Tuomas Rintamaki Mohammad Shoeybi Bryan Catanzaro
Wei Ping*,†

NVIDIA

2024.11.29

발제자 : 박현빈


phyeonbin01@gmail.com

What NVLM can do

(a)

the abstract

the paper



User Explain why this meme is funny.

NVLM-1.0

The humor in this image comes from the visual representation of the common phrase "the abstract and the paper." The phrase is often used to describe the difference between the summary of a research paper and the full paper itself. The image shows a large, fierce-looking lynx on the left, labeled "the abstract," and a small, domestic cat on the right, labeled "the paper." The contrast between the two animals is humorous because it visually represents the idea that the summary of a paper (the lynx) can be much more impressive or intimidating than the actual content of the paper (the cat).

What NVLM can do

(c)



User

You are a helpful driving assistant. In this scene, which lane should I choose and why?

NVLM-1.0

In this scene, you should choose the right lane. The signs indicate that the left two lanes are closed, while the right lane is open for buses and RVs. This suggests that the right lane is the only available option for vehicles, including buses and RVs.

(d)

- 1/ Set smallest number/minimum to first element (index 0) in the list.
- 2/ Look for the smallest number/minimum element in the list.
- 3/ Swap that value with item at index[min].
- 4/ Increment index of [min] to next element.
- 5/ Repeat until last element/list is sorted!

User

Write code based on the provided pseudo code.

NVLM-1.0

The pseudo code in the image describes the selection sort algorithm. The code for the same is as follows:

```
def selection_sort(arr):  
    n = len(arr)  
    for i in range(n):  
        min_index = i  
        for j in range(i+1, n):  
            if arr[j] < arr[min_index]:  
                min_index = j  
        arr[i], arr[min_index] = arr[min_index], arr[i]  
    return arr
```

기존 MLLM의 3가지 문제점

- MLLM들 간의 정확한 비교가 어렵다
 - 독점 모델(gpt-4o, Claude 3.5 Sonnet)에 대한 정보가 없다
 - 오픈소스 모델들도 서로 다른 backbone LLM, vision encoder, training data 사용
- 고차원 이미지를 사용했을 때 OCR task 성능은 오르지만, reasoning task(MMMU) 성능은 떨어진다
- 오픈소스 MLLM들은 multimodal task에서는 좋은 성능을 보여주지만, text-only 성능은 떨어진다

Contributions

- **Model Architecture**

- 동일한 LLM Backbone, vision encoder, training data를 사용한 모델 3개
- NVLM-D(decoder-only), NVLM-X(X-Attention), NVLM-H

- **High-resolution**

- Dynamic Tiling
- OCR 성능과 multimodal reasoning 성능 둘 다 향상

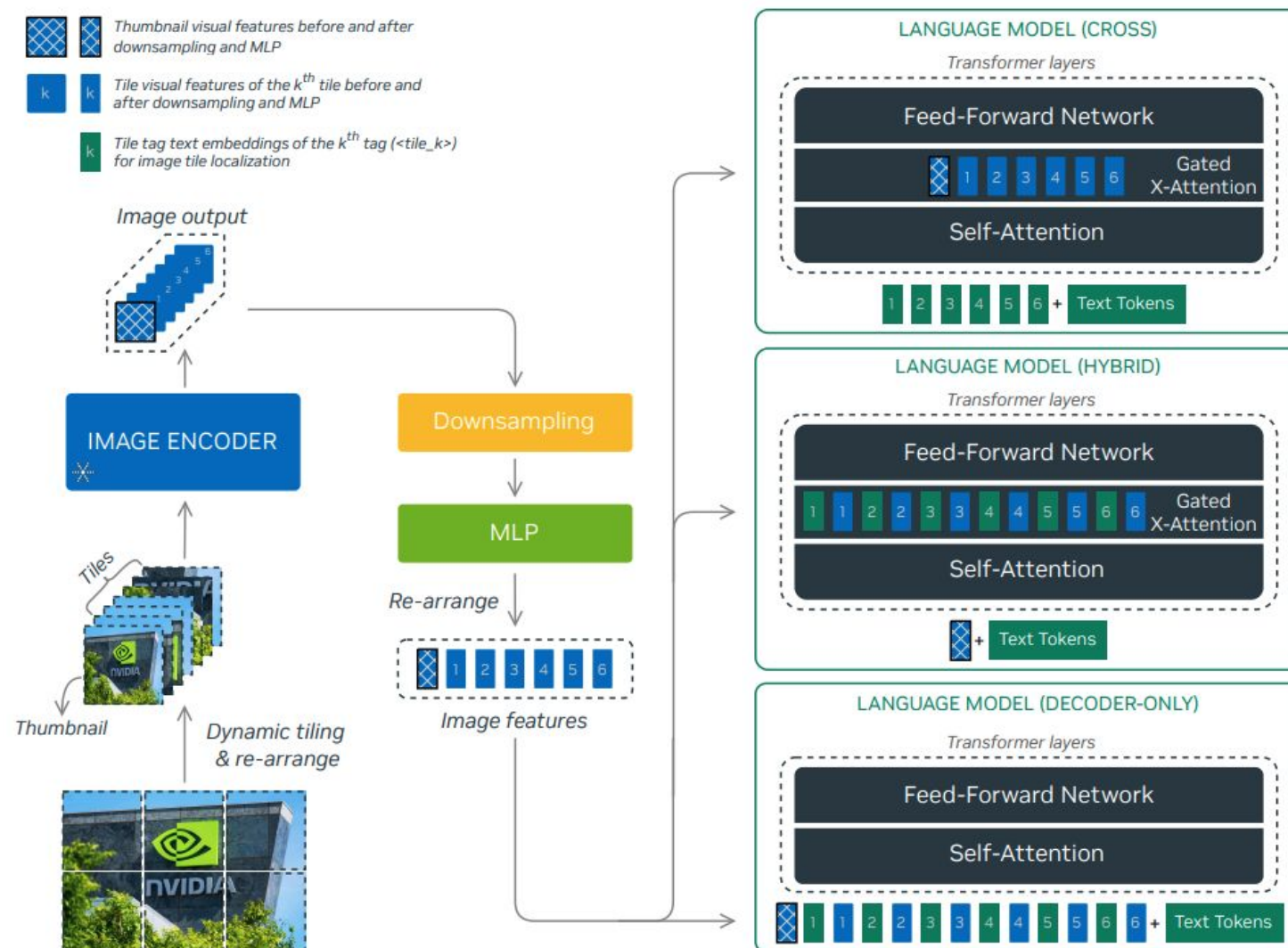
- **Training data**

- Pretraining dataset, SFT dataset 구축 및 공개

- **Production-grade multimodality**

- Vision-Language task, text-only task 둘 다 사용 가능

Model Architecture



Pre-trained LLM, Vision Encoder

- **LLM**

- 다양한 vision-language task를 잘 수행하기 위해 instruction-following 잘하는 모델 선택
- Backbone LLM : Qwen2-72B-Instruct
- For ablation study : Nous-Hermes-2-Yi-34B

- **Vision Encoder**

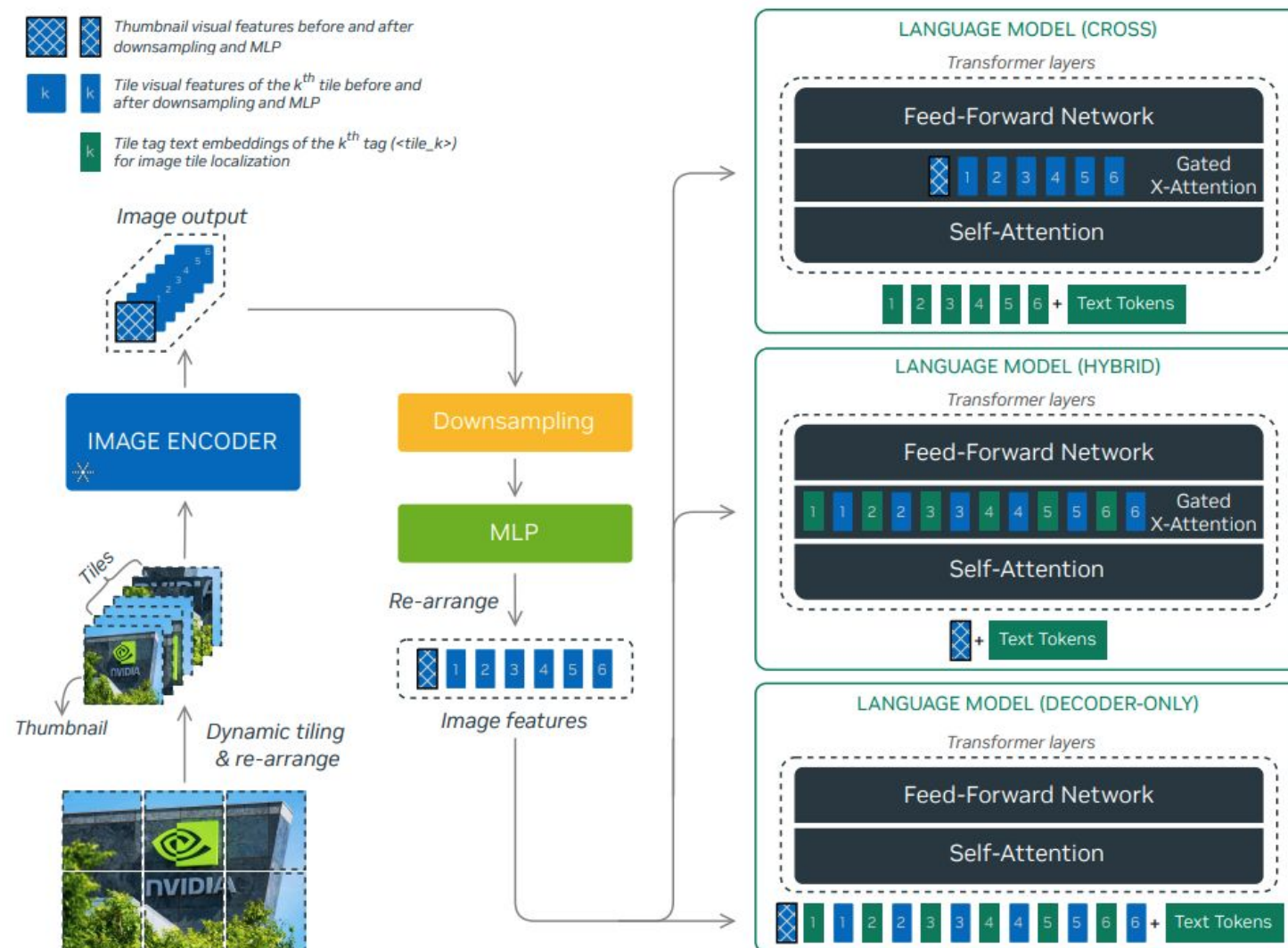
- InternViT-6B-448px-V1-5
 - 성능이 좋아 선택
 - vanilla ViT를 6B으로 scaling up 한 거
 - patch-size : 14
 - 448*448 이미지 처리

High-Resolution Inputs

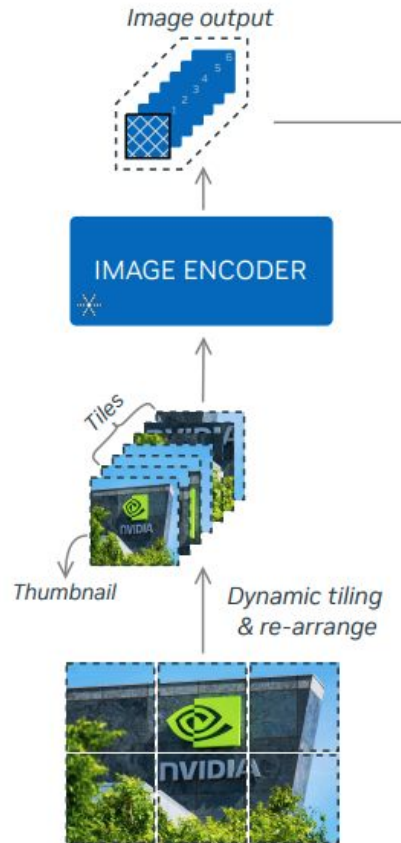
- OCR task에서 높은 성능을 달성하기 위해 고차원 이미지를 잘 다뤄야 한다
- 기존 Vision Encoder의 문제점
 - 고정된 낮은 해상도(224×224)로 사전 훈련
 - ViT-L/14에 224×224 이미지가 입력되면 $(224/14)^2 = 256$ 개의 토큰 생성
 - 높은 해상도(1024×1024)를 다루는 SAM encoder(ViT-L/16)도 존재
 - 하지만 input 이미지가 크든 작든 4,096개의 토큰 생성. High cost
- Dynamic high-resolution mechanism
 - 이미지를 압축하지 않고 여러 개의 tile로 분해
 - 각 타일을 Vision Encoder에 입력 후 토큰 생성
 - 672×448 해상도 이미지는 $(672/224) \times (448/224) = 6$ 개의 타일로 분해
 - 각 타일은 ViT-L/14를 통해 256개의 토큰으로 변경. 총 $256 \times 6 = 1,536$ 개의 토큰 생성



Model Architecture



Shared Vision Pathway



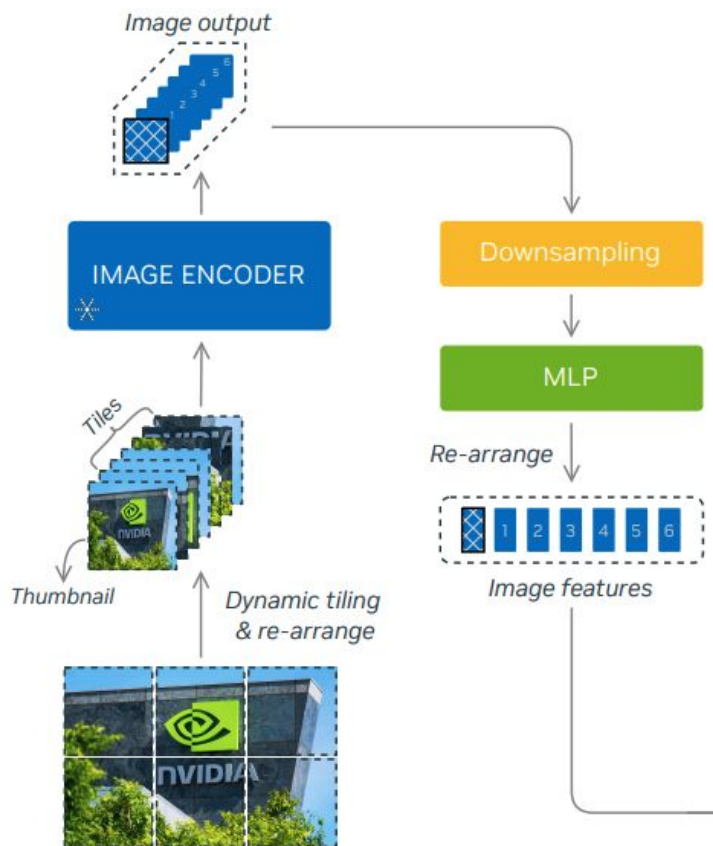
- **Input image 타일로 분해**

- 각 타일의 크기는 448×448
- 원본 사진을 448×448 로 압축하여 Thumbnail로 사용
- 타일의 개수 : n , 이때 $n \leq 7$

- **Image Encoder**

- InternViT-6B-448px-V1-5
 - ViT-L/14 기반
 - 각 타일은 $(448/14) \times (448/14) = 1,024$ 개의 토큰으로 변환
 - 각 토큰의 차원이 d 라고 할 때 각 타일은 $1024 \times d$.
($d=3,200$)
 - 따라서 최종 Image output의 크기는 $1,024 \times d \times n$.

Shared Vision Pathway



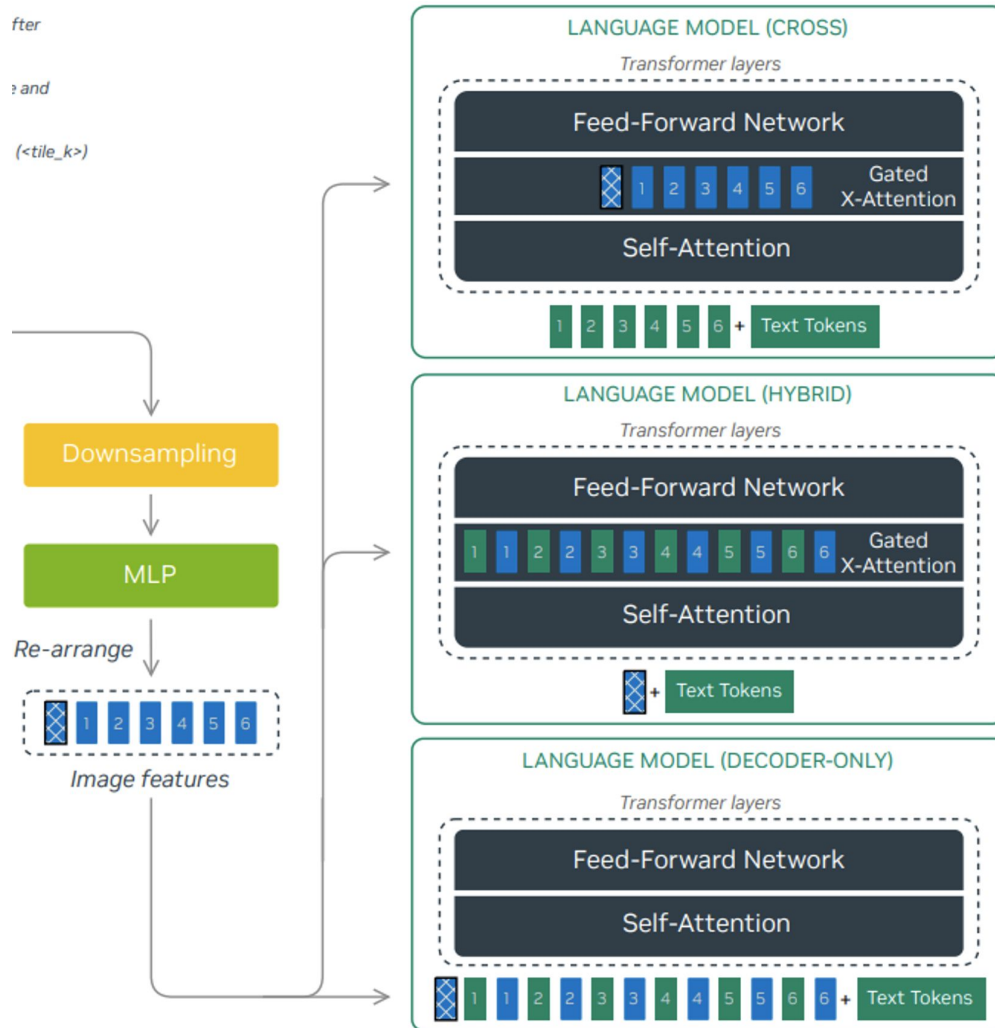
- **Downsampling**

- Attention 계산량 : n^2d . 토큰의 개수 n 을 줄여야 연산이 빨라짐
- 1024 -> 256 으로 다운샘플링
- $1024*d \rightarrow 256 * 4d$

- **MLP**

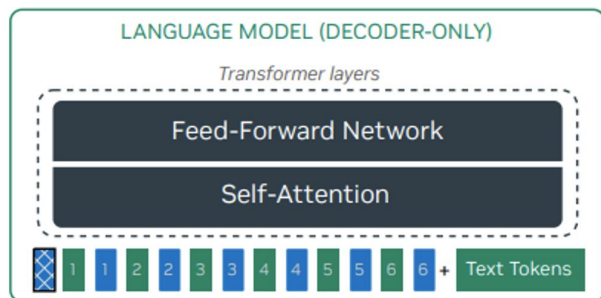
- image token의 차원을 text token의 차원과 같게 만드는 역할
 - 12,800 -> 8,192 or 7,168
- Random Initialized

Shared Vision Pathway



- **NVLM-D**
 - LLM에 image와 text 같이 입력
- **NVLM-X**
 - LLM에 text만 입력
 - Text를 Query, 이미지를 Key와 Value로 X-Attention
- **NVLM-H**
 - NVLM-D와 NVLM-X를 Hybrid한 모델
 - LLM에 Image thumbnail과 텍스트를 입력
 - Text를 Query, 이미지를 Key와 Value로 X-Attention

NVLM-D



- a) No tag: Simple concatenation without tile tag, which is the design of InternVL-1.5 [18].
- b) 1-D flattened tile tag: $\langle \text{tile_1} \rangle, \langle \text{tile_2} \rangle, \dots, \langle \text{tile_6} \rangle, \langle \text{tile_global} \rangle$.
- c) 2-D grid tag: $\langle \text{tile_x0_y0} \rangle, \langle \text{tile_x1_y0} \rangle, \dots, \langle \text{tile_xW_yH} \rangle, \langle \text{tile_global} \rangle$, where the $\{i : j\}$ of $\langle \text{tile_xi_yj} \rangle$ can be in $\{1:1, 1:2, 1:3, 1:4, 1:5, 1:6, 2:1, 2:2, 2:3, 3:1, 3:2, 4:1, 5:1, 6:1\}$.
- d) 2-D bounding-box tag: $\langle \text{box} \rangle (x_0, y_0), (x_1, y_1) \langle / \text{box} \rangle, \dots, \langle \text{box} \rangle (x_W, y_H), (x_{W+1}, y_{H+1}) \langle / \text{box} \rangle$, where the $(x_i, y_j), (x_{i+1}, y_{j+1})$ are the (left, top), (right, bottom) coordinates of that particular title within the whole high-resolution image.

Tile tag format	MMMU (val)	MathVista	AI2D (test)	ChartQA	DocVQA	TextVQA	OCRBench
Low-resolution (448^2)	50.9	46.1	67.0	64.8	52.9	78.2	622
DHR + No tag	50.0	51.7	79.9	76.1	80.2	78.4	728
DHR + 2-D grid tag	51.1	52.8	81.7	81.1	86.7	79.4	787
DHR + 2-D bbox tag	50.3	50.6	81.2	80.8	86.7	79.7	791
DHR + 1-D tag	52.0	53.8	82.1	81.1	87.4	79.9	806

- **Pre-training**

- 목적 : 2-layer MLP 학습
- LLM과 vision encoder frozen

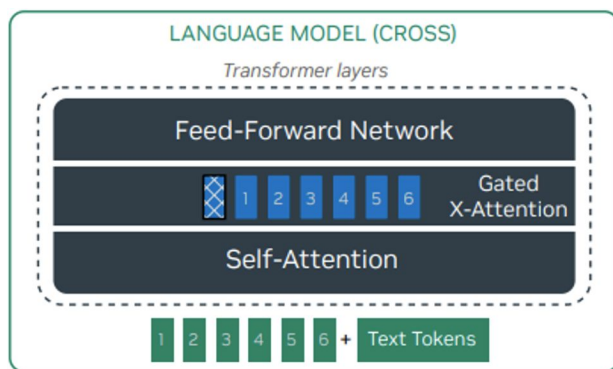
- **Supervised Fine-tuning**

- 목적 :
 - MLP 학습
 - vision 정보 습득을 위한 LLM 학습
- Vision encoder frozen

- **Tile Tag**

- LLM은 tile 구조에 대한 지식이 없다
- 따라서 이미지 타일 앞에 tile tag 부착
- 1-D는 2-D 정보를 가지고 있지 않아 일반화가 잘 됐기 때문에 성능이 좋다고 가정

NVLM-X



- **Pre-training**
 - 목적 : 1-layer MLP, X-Attention 학습
 - LLM과 vision encoder frozen
- **Supervised Fine-tuning**
 - Vision encoder frozen
- **Tile Tag**
 - NVLM-D와 동일한 tile tag 사용

Decoder-only vs X-Attention

Models	Batch size	# of H100 GPUs	Sequence length in LLM decoder	# of Tiles	Elapsed time (ms) per iteration	Throughput samples / sec
NVLM-X 34B	256	128	1,024	6+1	5,063	50.6
NVLM-D 34B	256	128	$1,024 + 256 \times 7 = 2,816$	6+1	8,885	28.8
NVLM-H 34B	256	128	$1,024 + 256 = 1,280$	6+1	7,071	36.2

- **Parameter Efficiency**

- NVLM-X는 X-Attention 때문에 파라미터가 더 많이 필요함

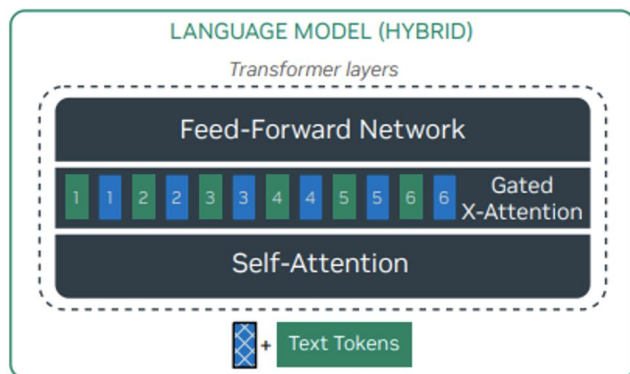
- **Training Efficiency**

- 처리량
 - NVLM-X : 50.6 samples/sec
 - NVLM-D : 28.8 samples/sec
 - Sequence length가 NVLM-D가 더 커서 처리량이 낮음

- **Multimodal reasoning**

- NVLM-D는 텍스트와 이미지를 같이 LLM에 주기 때문에 multimodal reasoning 성능이 좋지만, text token 대비 image token이 너무 많아지면 reasoning에 어려움을 겪을 수 있다

NVLM-H



- **NVLM-D처럼**
 - LLM에 image와 text를 같이 줘서 multimodal reasoning 성능을 높힘
- **NVLM-X처럼**
 - 입력 토큰 수를 줄여 처리량을 높힘
- **Tile Tag**
 - NVLM-D와 동일한 tile tag 사용

Multimodal PreTraining Data

- 목적
 - modality alignment module 학습 (MLP, X-Attention)
 - Open-source data
- 기존연구
 - X-Attention : 풍부하고 다양한 데이터셋
 - Decoder-only : 작더라도 좋은 품질의 데이터셋
- Finding
 - Decoder-only에서도 다양한 데이터가 중요

Pretraining data	MMMU (val)	MathVista	AI2D (test)	ChartQA	DocVQA	TextVQA	OCRBench
LLaVA-1.5 data [77]	51.8	48.9	80.5	80.3	85.2	78.9	760
Our pretraining data	52.0	53.8	82.1	81.1	87.4	79.9	806

Task	Dataset
Captioning	COCO [72], CC3M [127], SBU [114], LAION-115M (sanitized) [123; 66]
VQA (natural image)	VQAv2 [38], Visual Genome [59]
Chart	DVQA [51]
Document	Docmatix [90]
OCR / Scene-Text	OCR-VQA [98], COCO-Text [144], TextOCR [132], ReCTs [170], RRC-ArT [22], RRC-LSVT [134] RCTW [128], synthdog-en [57], pdfa-eng-wds [117]
Math	CLEVR-Math [73]

Multimodal SFT Data

- 목적
 - LLM이 텍스트와 이미지의 관계를 학습할 수 있도록
- 특징
 - 많은 OCR, Math 데이터셋
 - Text-only 데이터셋
 - LLM text-only 성능 하락 방지
 - 오픈소스 데이터 사용
 - 퀄리티를 높이기 위해 gpt-4o, gpt-4o-mini로 정제
- No Augmentation
 - Benchmark 성능은 오를 수 있지만
 - 일반화 성능에는 영향이 미미하기 때문
- Test
 - Not evaluated in a zero-shot setting
 - 독점 모델들의 벤치마크 결과도 zero-shot이 아니라는 가정
 - training과 test셋에 대한 정확도 차이

Task	Dataset
Captioning	COCO [72], TextCaps [130], ShareGPT-4o [61]
VQA (natural image)	VQAv2 [38], Visual Genome [59], TallyQA [2], Visual7W [177], Vizwiz [39]
General Knowledge	OK-VQA [91], A-OKVQA [125]
Visual Reasoning	GQA [45], Super-CLEVR [69], Raven [168], VSR [74]
Chart & Diagram	DVQA [51], PlotQA [97], MMC-Instruction [76], ChartQA [93], InfographicVQA [96], FigureQA [52], IconQA [84], Chart2Text [103], Diagram Image2Text [53]
Table	WikiTableQuestions [116], RobuT(WTQ, WikiSQL, SQA) [173], HiTab [20]
Document	DocVQA [95], Docmatix [90], DUDE [142], VisualMRC [135], TAT-DQA[175], UReader IE [160], UReader KG [160], UReader QA [160],
OCR / Screen / Scene-Text	OCR-VQA [98], TextVQA [131], ST-VQA [10], ScreenQA [43], SlideQA [136], PDF-VQA [29], VQA-CD [89], VQAonBD [1], POIE [60], SROIE [44], ORAND [28], EST-VQA [148], FUNSD [49], SQuAD(rendering) [121], WordArt [155], IAM [92], IIIT5K [46], HME100K [164], synthdog-en [57], Bentham QA [94], HW-SQuAD [94], WebSight [64], ChromeWriting [152], K12 Printing [65], COCO-Text [144], TextOCR [132], ReCTs [170], pdfa-eng-wds [117]
Math	CLEVR-Math [73], GeoQA+ [13], Geometry3K [83], TabMWP [86], GSM8K(rendering) [25], MetaMathQA(rendering) [162], MAVIS Data Engine [171], MAVIS Manual Collection [171], Geo170K Align [34], Geo170K QA [34], GeoMVerse [54], GEOS [126], UniGeo [14]
Science	AI2D [55], ScienceQA [85], TQA [56], ArXivQA [68], textbook data
Visual Instruction-Tuning	LRV-Instruction [75], LLaVA-158K [79], LLaVAR [172]
Text-only SFT	SlimOrca [70], ShareGPT [138], EvolInstruct [156], GPTTeacher [137], AlpacaGPT4 [118], UltraInteract [163], OrcaMathWordProblems [99], MathInstruct [165], MetaMath [162], GlaiveCodeAssistant [37], Magicoder [151], WizardCoder [88].

Vision-Language Benchmark

Tasks	MMMU test / val	MathVista testmini	VQAv2 test-dev	AI2D test / <i>no_mask</i>	TextVQA val	ChartQA test	DocVQA test	Real- WorldQA	OCR- Bench	Text-only Avg. 4
Proprietary										
GPT-4V [107]	56.1 / 56.8	49.9	77.2	78.2	78.0	78.5	88.4	61.4	645	-
GPT-4-Turbo [106]	- / 63.1	58.1	-	89.4	-	78.1	87.2	-	678	-
GPT-4o [108]	- / 69.1	63.8	-	94.2	-	85.7	92.8	-	736	-
Claude 3 Sonnet [5]	- / 53.1	47.9	-	88.7	-	81.1	89.5	51.9	646	-
Claude 3 Opus [5]	- / 59.4	50.5	-	88.1	-	80.8	89.3	49.8	694	-
Claude 3.5 Sonnet [6]	- / 68.3	67.7	-	94.7	-	90.8	95.2	-	788	-
Gemini Pro 1.0 [35]	- / 47.9	45.2	71.2	73.9	74.6	74.1	88.1	-	659	-
Gemini Ultra 1.0 [35]	- / 59.4	53.0	77.8	79.5	82.3	80.8	90.9	-	-	-
Gemini Pro 1.5 [36]	- / 58.5	52.1	80.2	80.3	73.5	81.3	86.5	67.5	-	-
Gemini Pro 1.5 (Aug 2024)	- / 62.2	63.9	80.2	94.4	78.7	87.2	93.1	70.4	754	-
Grok-1.5V [153]	- / 53.6	52.8	-	88.3	78.1	76.1	85.6	68.7	-	-
Grok-2 [154]	- / 66.1	69.0	-	-	-	-	93.6	-	-	-
Others										
QWen-VL-MAX	46.8 / 51.4	51.0	78.8	79.3	79.5	79.8	93.1	-	723	-
Adept Fuyu-Heavy [3]	- / 48.3	-	77.8	81.2	-	75.4	-	-	-	-
Open-access										
LLaVA-Next 34B [80]	44.7 / 51.1	46.5	-	-	69.5	-	-	-	574	-
VILA-1.5 40B [71]	46.9 / 51.9	-	84.3	-	-	-	-	-	-	-6.9
Cambrian-1 34B [139]	- / 49.7	53.2	-	79.7	76.7	75.6	75.5	67.8	600	-
LLaVA-OneVision 72B [65]	- / 56.8	67.5	-	85.6	-	83.7	91.3	-	-	-6.3
InternVL-1.2 40B [19]	- / 51.6	47.7	-	79.0	72.5	68.0	57.7	67.5	569	-
InternVL-1.5 26B [18]	- / 45.2	53.5	-	80.7	80.6	83.8	90.9	66.0	724	-
InternVL-2 40B [111]	- / 53.9	63.7	-	87.1	83.0	86.2	93.9	71.8	837	-
InternVL-2-Llama3-76B	- / 55.2	65.5	-	87.6 / 94.8	84.4	88.4	94.1	72.2	839	-6.7
*InternVL-2-Pro [111]	- / 58.9	66.3	-	87.3 / 96.0	-	87.1	95.1	-	837	-
*Llama 3-V 70B [32]	- / 60.6	-	79.1	93.0	83.4	83.2	92.2	-	-	0
*Llama 3-V 405B [32]	- / 64.5	-	80.2	94.1	84.8	85.8	92.6	-	-	0
NVLM-D _{1.0} 72B	54.6 / 59.7	65.2	85.4	85.2 / 94.2	82.1	86.0	92.6	69.7	853	+4.3
NVLM-X _{1.0} 72B	53.6 / 57.4	64.6	85.2	84.2 / 93.6	80.2	82.9	82.9	66.1	828	+2.5
NVLM-H _{1.0} 72B	53.0 / 60.2	66.6	85.2	83.8 / 93.3	80.3	83.3	83.1	66.0	831	+2.7

MMMU	대학 수준의 추론 문제
Math Vista	수학 능력
VQAv2	Image understanding
AI2D	초등학교 과학 문제
TextVQA	이미지에서 글자 읽기(OCR)
ChartQA	Chart understanding
DocVQA	VQA on document images
Real-WorldQA	Physical world perception and understanding
OCR-Bench	<ol style="list-style-type: none"> 이미지에서 글자 읽기 text 중심 VQA Document 중심 VAQ Information extraction Handwritten mathematical expression recognition

Text-only Benchmark

Tasks	Backbone LLM	MMLU	GSM8K	MATH	HumanEval	Avg. Accuracy	Text-only Avg. 4
Proprietary							
GPT-4o [108]	N/A	88.7	-	76.6	90.2	-	unknown
Gemini Pro 1.5 (Aug 2024) [36]	N/A	85.9	90.8	67.7	84.1	82.1	unknown
Claude 3.5 Sonnet [6]	N/A	88.7	96.4	71.1	92.0	87.0	unknown
Open LLM							
(a) Nous-Hermes-2-Yi-34B [102]	N/A	75.5	78.6	21.8	43.3	54.8	N/A
(b) Qwen2-72B-Instruct [119]	N/A	82.3	91.1	<u>59.7</u>	86.0	<u>79.8</u>	N/A
(c) Llama-3-70B-Instruct [32]	N/A	82.0	93.0	51.0	81.7	<u>76.6</u>	N/A
(d) Llama-3.1-70B-Instruct [32]	N/A	83.6	95.1	68.0	80.5	81.8	N/A
(e) Llama-3.1-405B-Instruct [32]	N/A	87.3	96.8	73.8	89.0	86.7	N/A
Open Multimodal LLM							
VILA-1.5 40B [71]	(a)	73.3	67.5	16.8	34.1	47.9	- 6.9
LLaVA-OneVision 72B [80]	(b)	80.6	89.9	49.2	74.4	73.5	- 6.3
InternVL-2-Llama3-76B [111]	(c)	78.5	87.1	42.5	71.3	69.9	- 6.7
*Llama 3-V 70B [32]	(d)	83.6	95.1	68.0	80.5	81.8	0
*Llama 3-V 405B [32]	(e)	87.3	96.8	73.8	89.0	86.7	0
NVLM-D _{1.0} 72B	(b)	82.0	92.9	<u>73.1</u>	88.4	<u>84.1</u>	+ 4.3
NVLM-X _{1.0} 72B	(b)	81.4	91.8	<u>70.6</u>	85.2	<u>82.3</u>	+ 2.5
NVLM-H _{1.0} 72B	(b)	80.4	91.5	<u>71.4</u>	86.6	<u>82.5</u>	+ 2.7

MMLU	57개의 주제(역사, 법, 컴퓨터공학) 다지선다
GSM8K	초등학생 수학 문제
MATH	다양한 난이도의 수학 문제
HumanEval	코딩

Frozen vs Unfrozen during Multimodal SFT

- **Freezing**
 - LLM의 text 성능 하락이 없다
 - 멀티모달 성능 하락
- **Unfreezing**
 - LLM의 text 성능 하락이 있을 수 있지만 NVLM은 text-only dataset을 사용해 이를 방지했고, 오히려 text 성능이 향상됐다
 - 멀티모달 성능 향상

Tasks	MMMU test / val	MathVista testmini	VQAv2 test-dev	AI2D test	TextVQA val	ChartQA test	DocVQA test	RealWorld- QA	OCR- Bench
NVLM-X 34B (frozen)	43.2 / 51.6	51.8	83.8	72.4	72.4	74.4	73.2	63.4	696
NVLM-X 34B	47.2 / 54.0	59.2	84.5	79.6	78.2	79.4	79.2	64.8	802
NVLM-X 72B (frozen)	50.6 / 54.4	60.6	85.3	76.2	76.2	76.2	76.4	65.3	722
NVLM-X _{1.0} 72B	53.6 / 57.4	64.6	85.2	84.2	80.2	82.9	82.9	66.1	828

느낀점

장점

1. 기존 MLLM들은 Backbone LLM과 사용된 데이터셋이 다 달라 MLLM들간 정확한 비교가 어려웠지만 NVLM은 같은 LLM과 데이터셋을 사용한 모델 3가지가 있기 때문에 직접적인 비교가 가능하다
2. Dynamic tiling을 통해 고해상도 이미지를 처리하여 OCR에서 높은 성능을 보이며, 동시에 Multimodal reasoning에서도 좋은 성능을 보인다
3. Text-only 데이터셋을 통해 multimodal 학습으로 인한 LLM의 text 성능 감소를 방지한다
4. Open source로 공개

단점

1. Text-only 데이터셋을 통해 LLM의 text 성능이 올랐다고 하는데, 성능 측정에 사용된 벤치마크가 NVLM에 유리한 방향으로 편향되어 있다

Open Question

LLM을 Frozen 시키고, Multimodal data를 더 많이 줘도 괜찮지 않을까?

text 성능은 유지가 되고, Multimodal 성능은 향상되지 않을까?