

TinyLLaVA: A Framework of Small-scale Large Multimodal Models

this paper focuses on designing and analyzing
small-scale LMMs

Wider context before jumping into details of the paper

Similar model

<https://huggingface.co/spaces/HuggingFaceM4/idefics-8b>

<https://huggingface.co/spaces/qnguyen3/nanoLLaVA>

What is a LMM (Large multiModal Model)

- Beyond Single-Modality Limitations:
 - Theory, more modalities the better
 - Holistic and contextually-aware approach to information processing.
- Takes in one or more types of data and transforms to another.
- Understanding Through Fusion:
 - Don't just analyze each type of data independently.
 - Learn relationships between modalities and merge insights in a process called "fusion".
 - Provides richer and nuanced understanding of the information.
- Using adapters vs training from scratch
 - This paper discusses using adaptors
 - From scratch potentially better
 - Different designs
 - More time to converge modalities

Why did I want to read this paper

- It introduces recipes or methods to cook your own multimodal models
- Uses small models for quicker responses, specific applications.
- The power of mixing modalities
 - Humans can listen, see, touch, smell, taste
 - A passion of mine is IOT, sensing the physical world.
 - So many sensors
- Hugging face has 10,000's of trained models to use as ingredients
 - Specialised for specific purposes
 - General purpose
- GPU resources to fuse models low VS training from scratch.

Why small when most people go large

- Small can perform as well as Large models with:
 - The right data
 - The right model
 - and/or more targeted to a domain.
- Helps the GPU poor run models fit for purpose
- Cheaper inference cost
- Faster response rates
- Serve more people on less(er) hardware

A bit about the models used




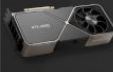





- **Language models**
 - [TinyLlama \(1.1B\)](#)
 - pretrain on 3 trillion tokens
 - 90 days using 16 A100-40G GPUs - to buy £ 320,000
 - [StableLM-2-1.6B\(1.6B\)](#)
 - pre-trained on 2 trillion tokens
 - [Phi-2\(2.7B\)](#)
 - Training tokens 1.4T tokens
 - GPUs: 96xA100-80G - to buy £1,920,000
 - Training time: 14 days
- **Clip models (shared **vision and text** model that outputs shared vectors aka embeddings)**
 - [CLIP\(428M\)](#)
 - ViT-L/14 Transformer architecture as an image encoder and uses a masked self-attention Transformer as a text encoder
 - [SigLIP \(878M\)](#)
 - A SoViT Shape optimized Vision Transformer optimized for both width and depth, as well as the MLP size, achieves results that are competitive with larger models.
 - SOTA performance on, image classification, captioning, VQA, and zero-shot transfer, effective across a broad range of domains

Model Size to VRAM estimate

To give a idea of vram size and what type of GPU they fit on

- 32bit precision (parameter size)
 - TinyLlama (1.1B): around 12GB of VRAM. Optimal training, 16GB+.
 - StableLM-2-1.6B(1.6B): 16GB of VRAM minimum, 24GB+
 - Phi-2(2.7B): Professional-grade GPUs 32GBVRAM+
- After 4-bit quantization (parameter size)
 - TinyLlama (1.1B): Potentially could fit within 4-6GB of VRAM.
 - StableLM-2-1.6B(1.6B): Might be possible to fit within 6-8GB of VRAM.
 - Phi-2(2.7B): Likely to still require around 8-12GB of VRAM

Graphics cards

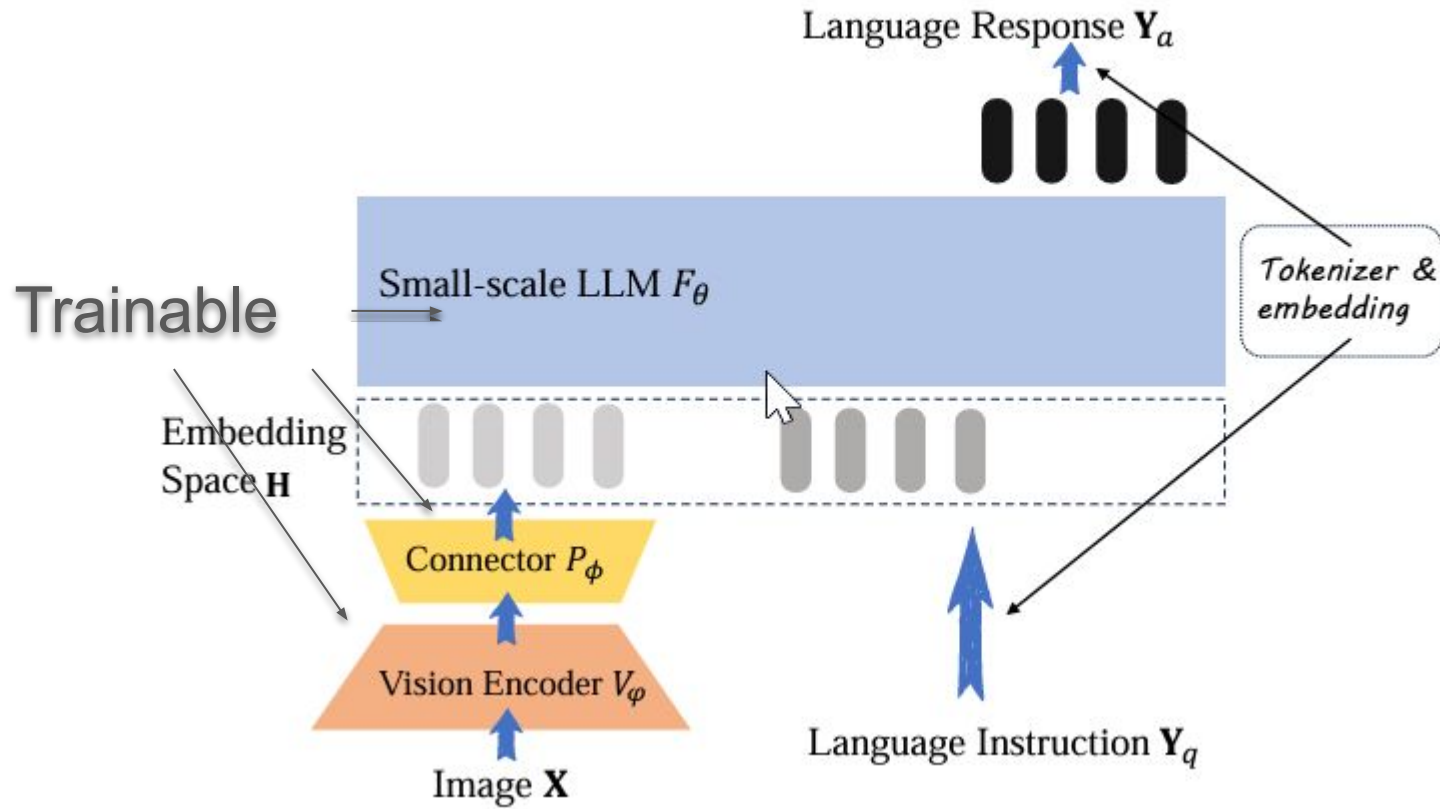
GPU Specifications									
	RTX 4090	RTX 4080	RTX 4070 Ti	RTX 3090	RTX 3070	RTX A6000	TITAN RTX	RTX 2080 Ti	Quadro RTX 8000
GPU Architecture	 Ada Lovelace AD-102	 Ada Lovelace AD-103	 Ada Lovelace AD-104	 Ampere GA-102	 Ampere GA-104	 Ampere GA-102	 Turing TU-102	 Turing TU-102	 Turing TU-102
Clock Speed Base (MHz)	2235	2205	2310	1395	1500	1410	1350	1350	1395
Clock Speed Boost (MHz)	2520	2505	2730	1695	1725	1800	1770	1545	1770
Memory Size	24GB	16GB	12GB	24GB	8GB	48GB	24GB	11GB	48GB
Memory Type	GDDR6X	GDDR6X	GDDR6X	GDDR6X	GDDR6	GDDR6	GDDR6	GDDR6	GDDR6
Memory Bandwidth	1008 GB/s	717 GB/s	504 GB/s	936 GB/s	448 GB/s	768 GB/s	672 GB/s	616 GB/s	672 GB/s
CUDA Cores	16384	9728	7680	10496	5888	10752	4608	4352	4608
Tensor Cores	512 4th Gen	304 4th Gen	240 4th Gen	328 3rd Gen	184 3rd Gen	336 3rd Gen	576 2nd Gen	544 2nd Gen	576 2nd Gen
RT Cores	128 3rd Gen	76 3rd Gen	60 3rd Gen	82 2nd Gen	46 2nd Gen	84 2nd Gen	72 1st Gen	68 1st Gen	72 1st Gen
Compute Pwr FP32 GFLOPS	82580	48740	41930	35580	20310	38710	16310	13450	16310
Compute Pwr FP64 GFLOPS	1290	762	655	556	317	1209	510	420	510
TDP	450W	320W	285W	350W	220W	300W	280W	260W	295W

Language models vs the Clip models

- Language Model
 - Trained on language tokens only
 - Language in and language out
 - Larger parameter sizes
- Clip Models (CLIP and SigLIP) aka Vision-Language Models
 - Language and image in and language out
 - Smaller parameter sizes
 - CLIP and SigLIP pre-trained on large datasets of image-text pairs to learn a shared embedding space.
 - CLIP uses a contrastive loss function, which serves as a defining characteristic.

The TinyLLava framework

Model architecture



The vision encoder

- Take in an image and output text embedding
 - Trained to maximize the similarity of (image, text) pairs
- 2 types of encoders tested
 - CLIP (Contrastive Language-Image Pre-training)
 - SigLP (shape-optimized model)
- Resolution input
 - SigLP accepts higher res images
- Vision tokens outputs
 - 729 for SigLP
 - 576 for CLIP
- Both transformer based like attention paper but..
 - vision transformer (ViT)
 - Encoder only
 - Learns to associate text with matching images and disassociate text from non-matching images

The connector

- Purpose: Enables the language model to see
- The connector can be thought of as a translator
 - Bridges the vision model to the language model
 - Converts the patch vectors into vectors that the language model understands
- Multi-layer perceptron

The LLM

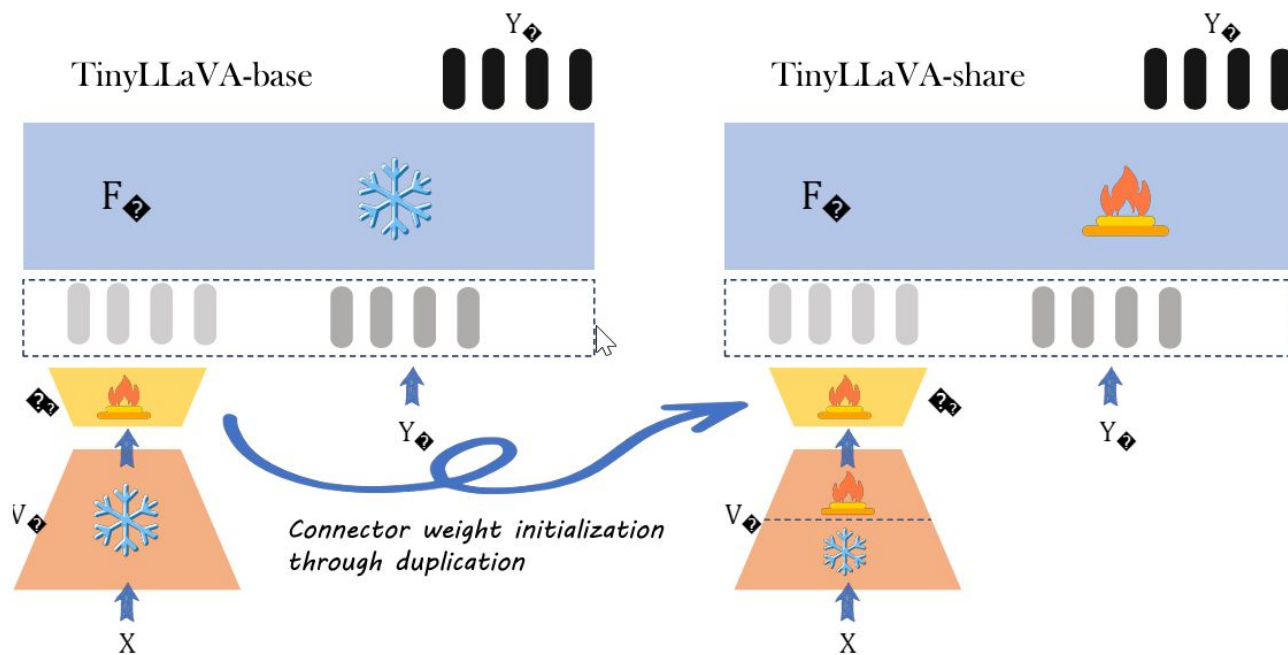
- Takes in visual and text embeddings and outputs text
- Each of the models TinyLlama (1.1B), StableLM-2-1.6B (1.6B), and Phi-2 (2.7B) - are transformer-based language models.

The Training pipeline

- Pre-training for Feature Alignment first -> Supervised Fine-tuning last

Aspect	Pre-training for Feature Alignment	Supervised Fine-tuning
Focus	Aligning different modalities (vision and text) within a shared embedding space.	Optimizing the model's response accuracy in a conversational context.
Data Format	Uses image-caption style data (image and corresponding descriptive response).	Utilizes the full format of multi-turn conversations.
Training Objective	Establishes a broad understanding across modalities.	Aims to perfect the model's performance in generating appropriate responses.

Base recipe vs Share recipe



In the **base recipe**, keep parameters of both the vision encoder and small-scale LLM frozen and solely updating the connector.

In the **share recipe**, freeze the first 12 layers of the vision encoder and update the rest of the model. Additionally, initialize connector from the base's pretrained counterpart.

The experiment setup

Training datasets

- [ShareGPT4V](#)
 - Images passed through GPT4 and trained
- LLaVA-1.5
 - [LLaVA-Instruct-150K](#)

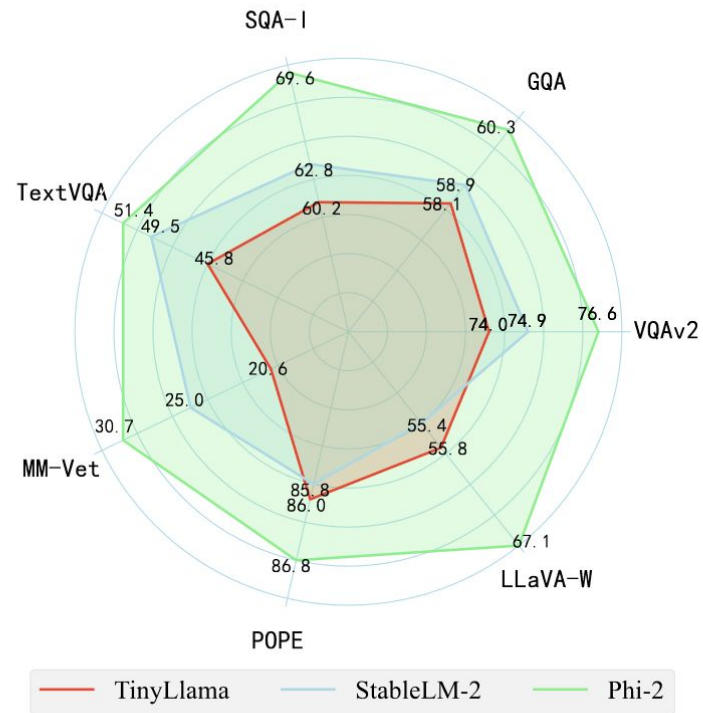
Training recipes

- 1st recipe (LLaVA-1.5)
 - Pre training
 - Vision encoder and small-scale LLM frozen
 - Connector updated
 - supervised fine-tuning
 - The vision encoder remains frozen
 - Connector and small-scale LLM updated
 - tune the model for one epoch with a learning rate of $2e-5$ and a batch size of 128
- 2nd recipe (ShareGPTv4)
 - Pre-training
 - The connector is initialized from the base recipe's pre-trained counterpart (LLaVA-1.5)
 - Vision encoder is partially updated
 - Supervised fine-tuning
 - The vision encoder remains frozen
 - Connector and small-scale LLM are updated.
 - update the rest of the model for one epoch with learning rate of $2e-5$ and a batch size of 256

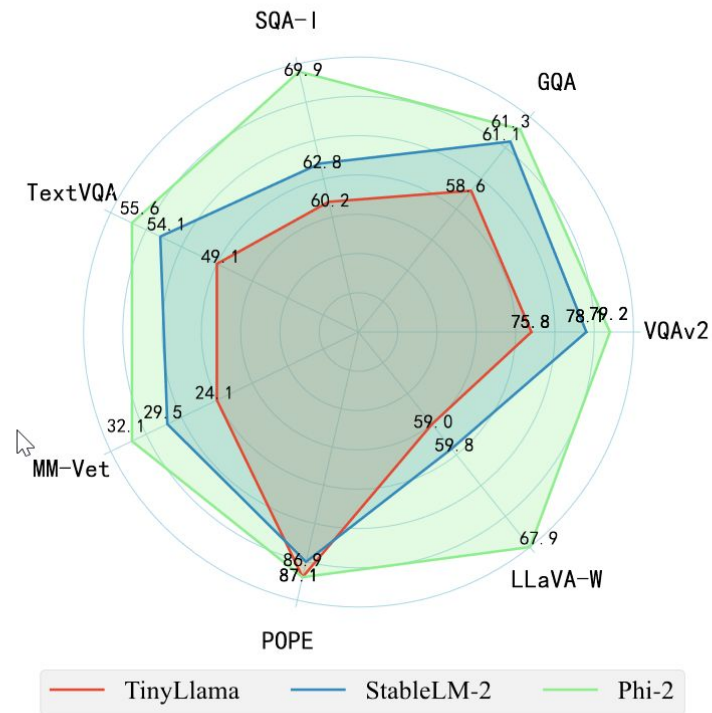
Evaluation Benchmarks

Dataset/Benchmark	Description	Type
VQAv2	A dataset with images from COCO, designed to evaluate a model's visual recognition, grounding, spatial reasoning, and language understanding abilities.	Visual & Language Reasoning
GQA	Focuses on testing a model's ability to perform visual and compositional reasoning, using data aligned with the Visual Genome dataset.	Visual & Compositional Reasoning
TextVQA	Tests models on recognizing and reasoning over textual information embedded within images.	Textual Recognition in Images
ScienceQA-IMG	A subset of ScienceQA, this benchmark uses images to test a model's ability to reason with scientific knowledge.	Scientific Reasoning
POPE	Assesses if a language model hallucinates (fabricates information), requiring it to determine whether objects within prompts exist or not.	Evaluation of Hallucination
MM-Vet	A comprehensive evaluation benchmark for LMMs, testing them on visual recognition, spatial reasoning, general knowledge, language generation, visual math, and OCR tasks.	Multi-Modal Evaluation
LLaVA-W	Evaluates LLM performance on challenging tasks and tests their ability to generalize knowledge to new domains.	LLM Generalization
MME	A benchmark for measuring both perception and cognitive abilities of LMMs across 14 different subtasks.	Perception & Cognitive Abilities
MMBench	Comprehensively assesses LLM capabilities across a broad set of 20 dimensions.	Broad LLM Capabilities

The language and vision model ablation tests

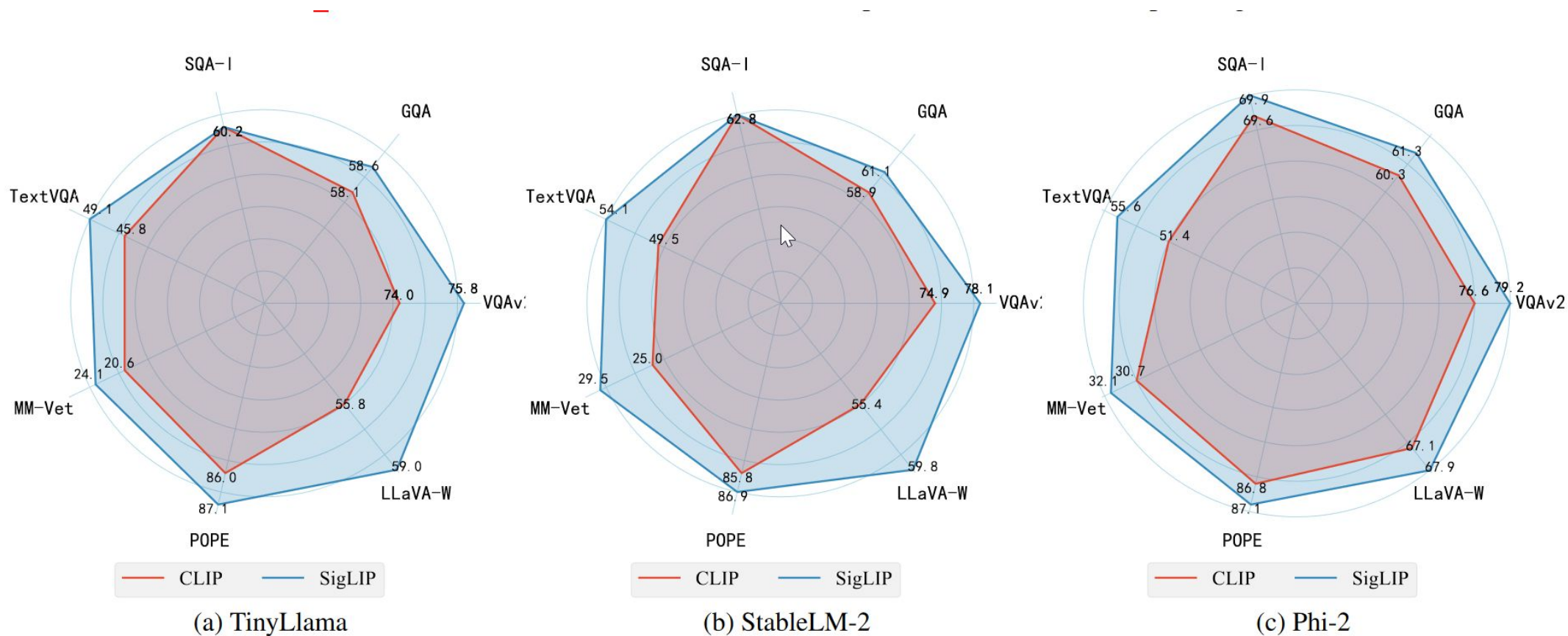


(a) CLIP

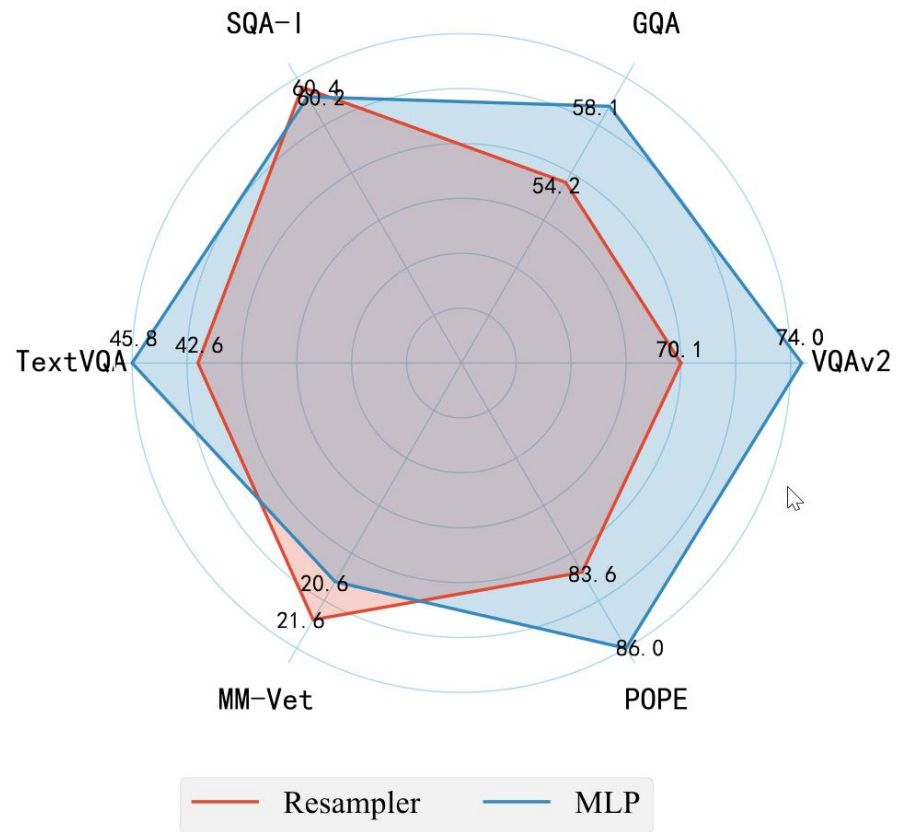


(b) SigLIP

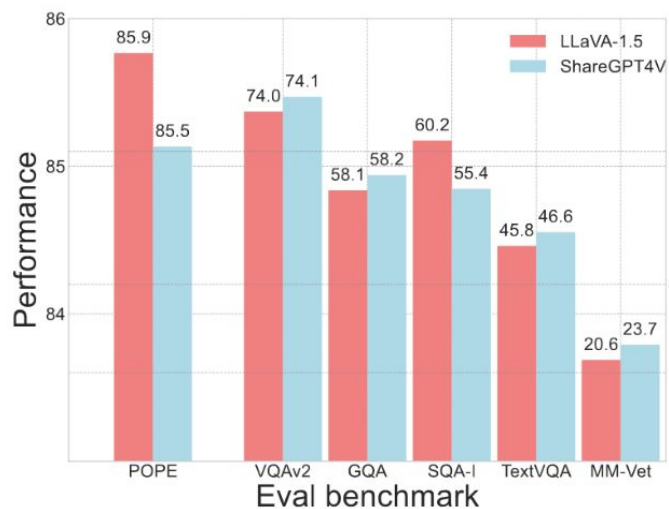
The vision and language model ablation tests



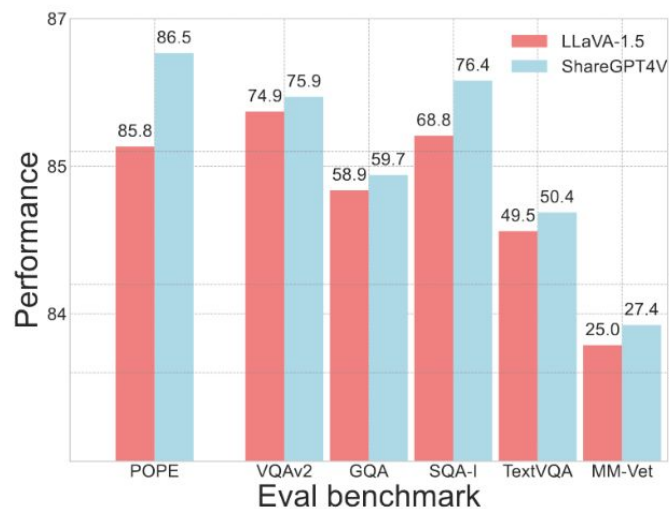
Mlp vs sampler



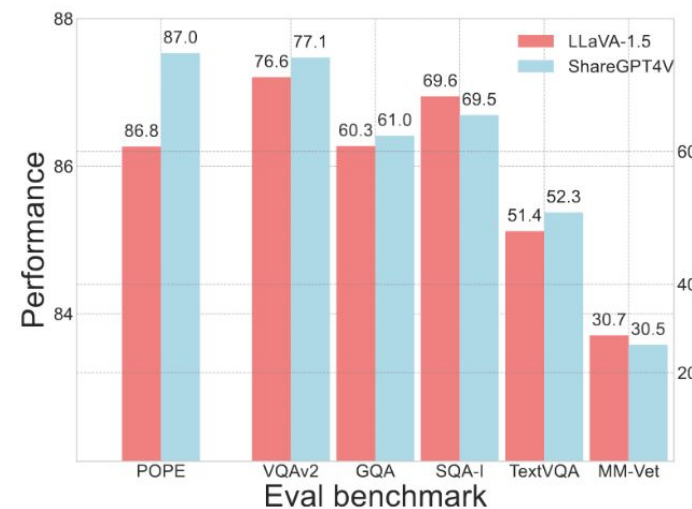
Language model vs Training datasets ablation



(a) TinyLlama

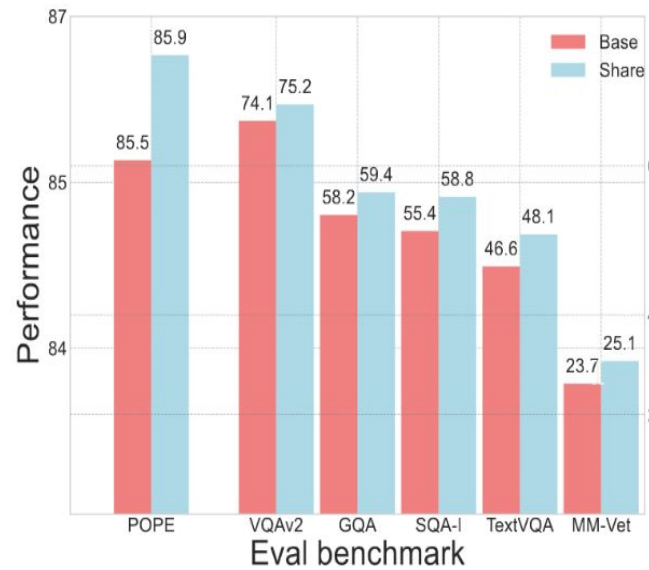


(b) StableLM-2

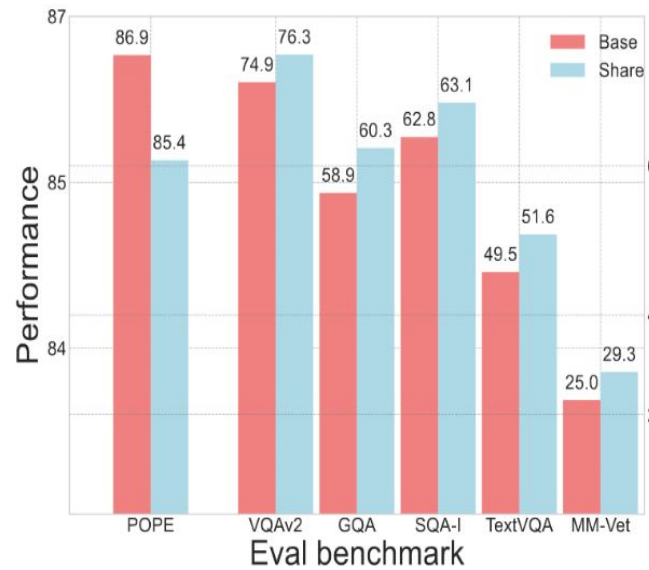


(c) Phi-2

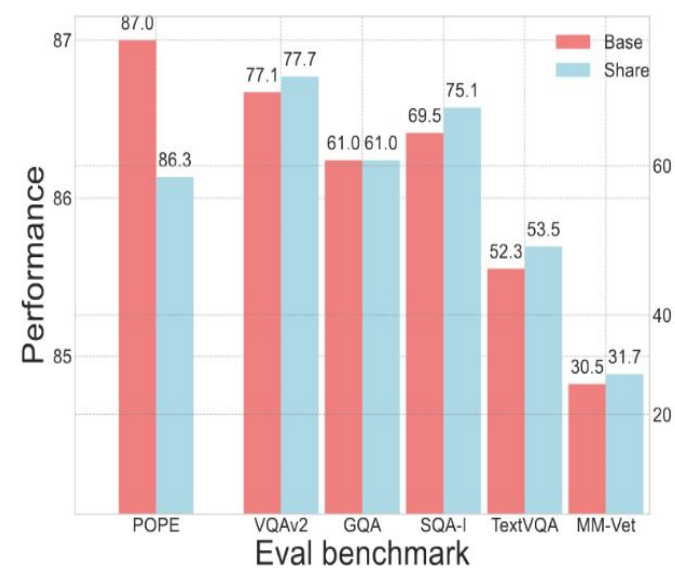
Training recipes vs language model ablations



(a) TinyLlama

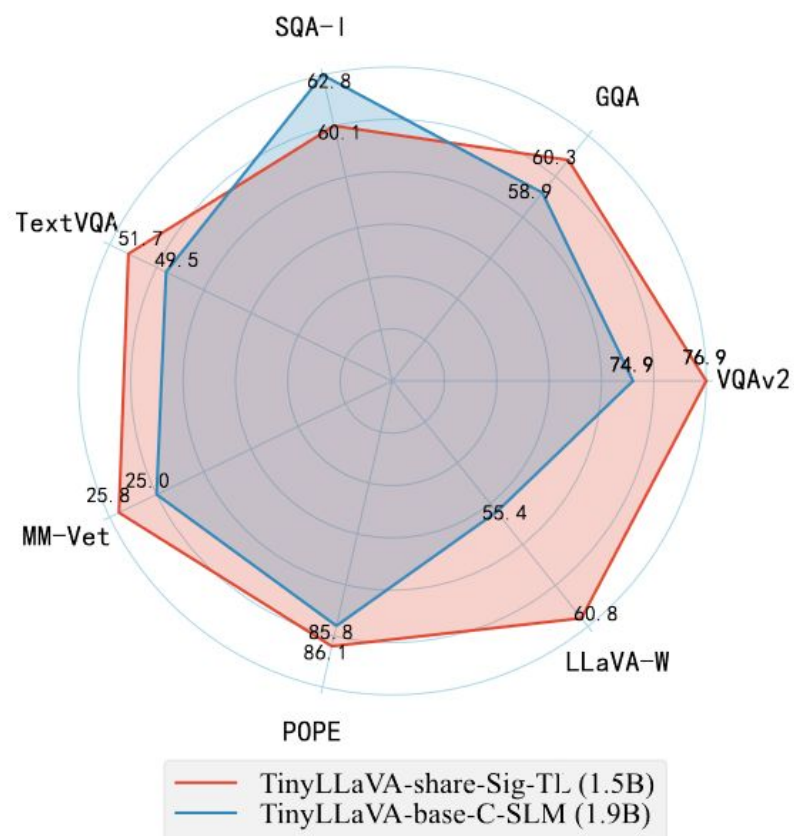


(b) StableLM-2

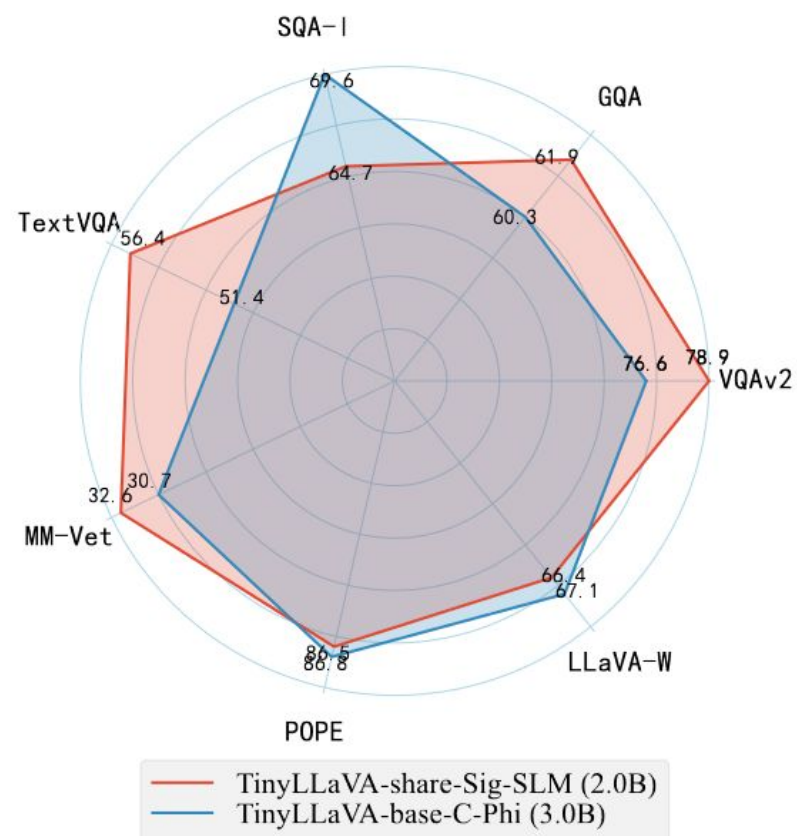


(c) Phi-2

TinyLLaVAs VS other models



(a)



(b)

The code and how to use

- <https://github.com/DLCV-BUAA/TinyLLaVABench>
 - [2024.03.10] Finetune scripts out!

TinyLLaVA Competition

- Similar but more popular
 - <https://github.com/vikhyat/moondream>
 - The website <https://moondream.ai/>
- Larger model that doesn't use a adaptor
 - <https://publications.reka.ai/reka-core-tech-report.pdf>