

## **1. FOUNDATIONS OF MLLMs**

Evolution of LLMs to multimodal models; architectures, training, and alignment

## **2. MULTIMODAL REASONING**

Datasets, benchmarks, and techniques for reasoning over visual documents

## **3. HUMAN-AI INTERACTION**

Multimodal agents, GUI grounding, and interactive data analysis.

## **4. RESPONSIBLE & INCLUSIVE AI**

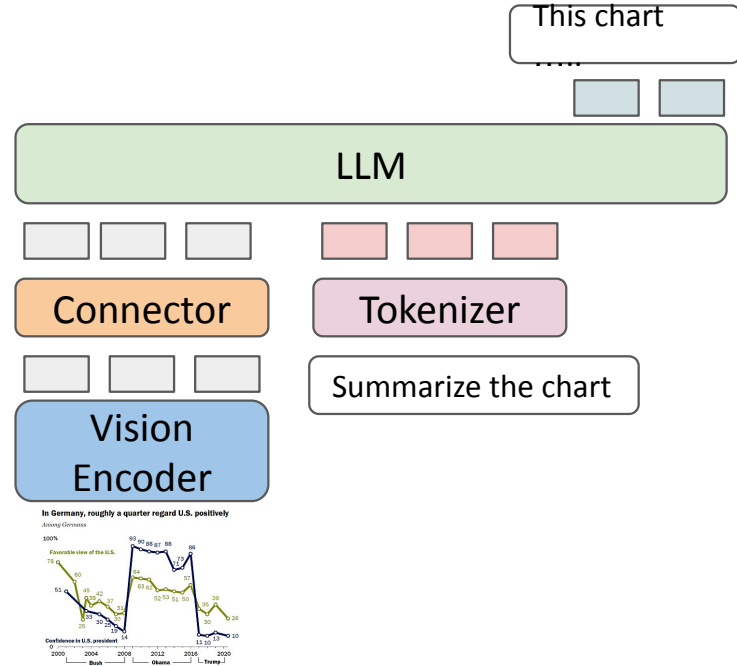
Accessibility, multilingual understanding, fairness, and hallucination risks

## **Future Challenges & Outlook**

# Architecture: Vision-Language Models

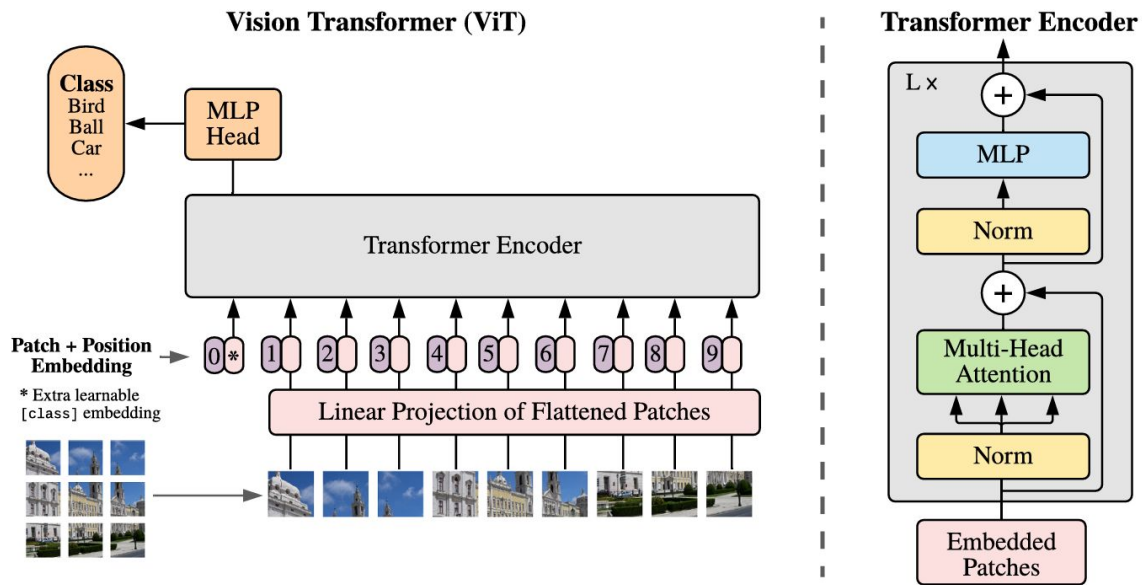
Vision Language Models (VLMs) typically consist of **three** components:

1. **Vision Encoder**: pretrained on images.
2. **LLM**: pretrained on text.
3. **Connector** that maps visual features into the LLM's text space.



# Architecture: Vision Encoders

*How do vision transformers encode the image?*



An Image is worth 16x16 words ([Dosovitskiy et al. 2020](#))

# Architecture: Vision Encoders

*How to pretrain the ViT?*

## 1. Self-supervised Pretraining

a. *Pretrains on images only*



(Caron et al. 2021)

## 2. Language-supervision

a. *Pretrains on Image-text pairs*

## 3. Unified Recipes



Pepper the  
aussie pup

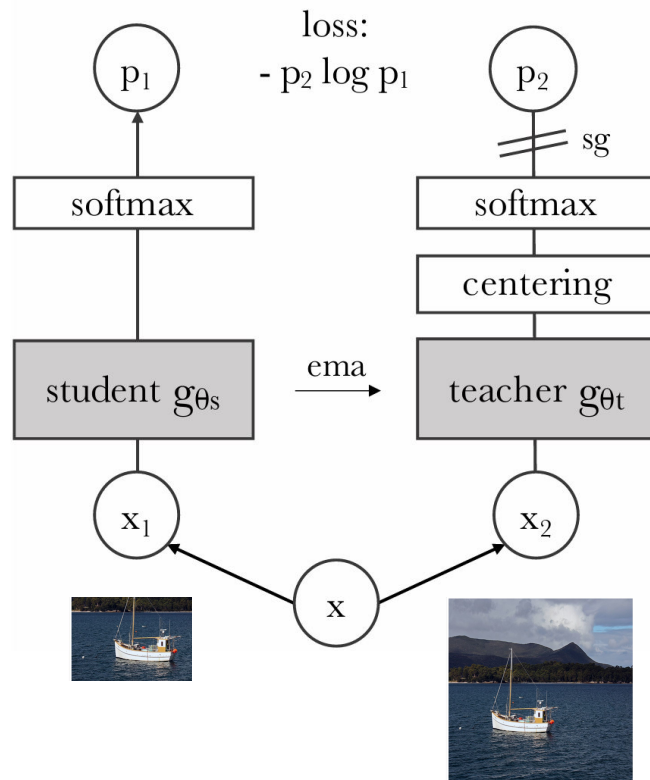
(Radford et al. 2021)

# Architecture: Vision Encoders

How to pretrain the ViT?

## 1. Self-supervised Pretraining (DINO)

- Learns visual features by matching different views of the same image (no labels)
- Optimizes cross-entropy between teacher and student outputs
- Produces **invariant representations** useful for vision-centric downstream tasks

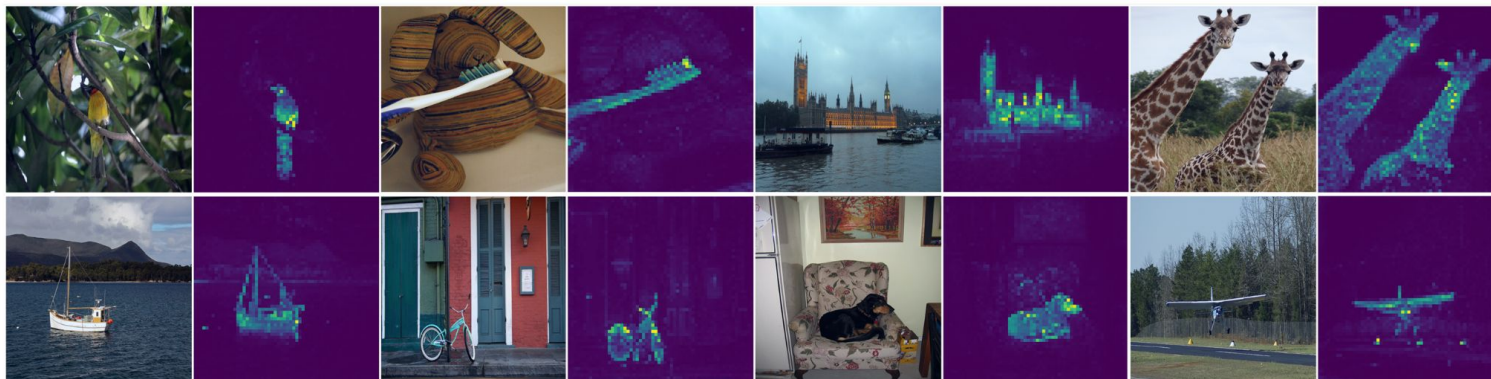


(Caron et al. 2021)

# Architecture: Vision Encoders

## 1. Self-supervised Pretraining (DINO)

- a. The model learns **low-level dense features** without any supervisions
- b. Useful for **vision-centric tasks** such as **object detection** and **segmentation**



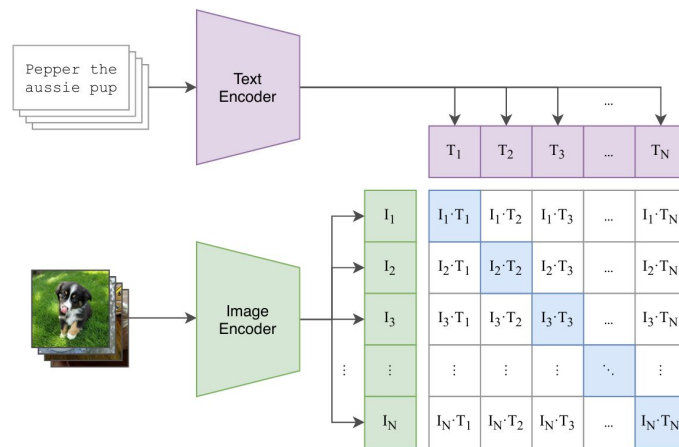
# Architecture: Vision Encoders

## 2. Language Supervision

### 1. CLIP (Contrastive Language–Image Pretraining)

- a. Learn aligned embeddings for images and text using paired data
- b. Train with contrastive loss:
  - i. match correct image-text pairs
  - ii. push apart mismatched ones
- c. The model learns high-level semantics (object categories, text)

$$-\frac{1}{2|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \left( \overbrace{\log \frac{e^{t\mathbf{x}_i \cdot \mathbf{y}_i}}{\sum_{j=1}^{|\mathcal{B}|} e^{t\mathbf{x}_i \cdot \mathbf{y}_j}}}_{\text{image} \rightarrow \text{text softmax}} + \overbrace{\log \frac{e^{t\mathbf{x}_i \cdot \mathbf{y}_i}}{\sum_{j=1}^{|\mathcal{B}|} e^{t\mathbf{x}_j \cdot \mathbf{y}_i}}}_{\text{text} \rightarrow \text{image softmax}} \right)$$



(Radford et al. 2021)

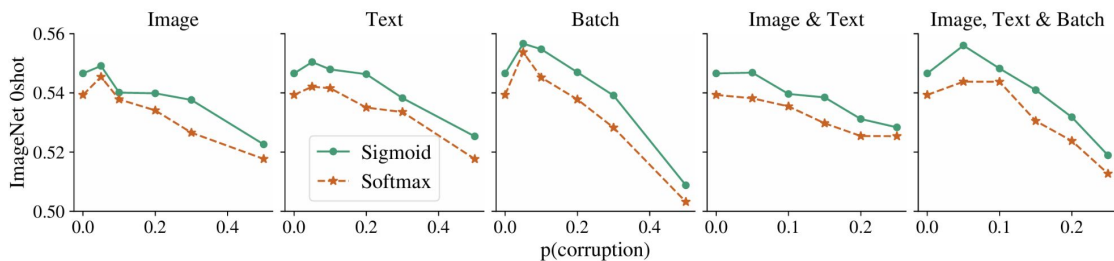
# Architecture Principles: Vision Encoders

## 2. Language Supervision

### 2. SigLIP (Sigmoid Loss)

- Replaces softmax contrastive loss with **sigmoid (binary) loss**
- Treats each image-text pair as **positive or negative independently** (no global normalization)
- More scalable training to larger batch size.
- More robust to noise and data corruption.

$$-\frac{1}{|\mathcal{B}|} \sum_{i=1}^{|\mathcal{B}|} \sum_{j=1}^{|\mathcal{B}|} \underbrace{\log \frac{1}{1 + e^{z_{ij}(-\mathbf{x}_i \cdot \mathbf{y}_j + b)}}}_{\mathcal{L}_{ij}}$$

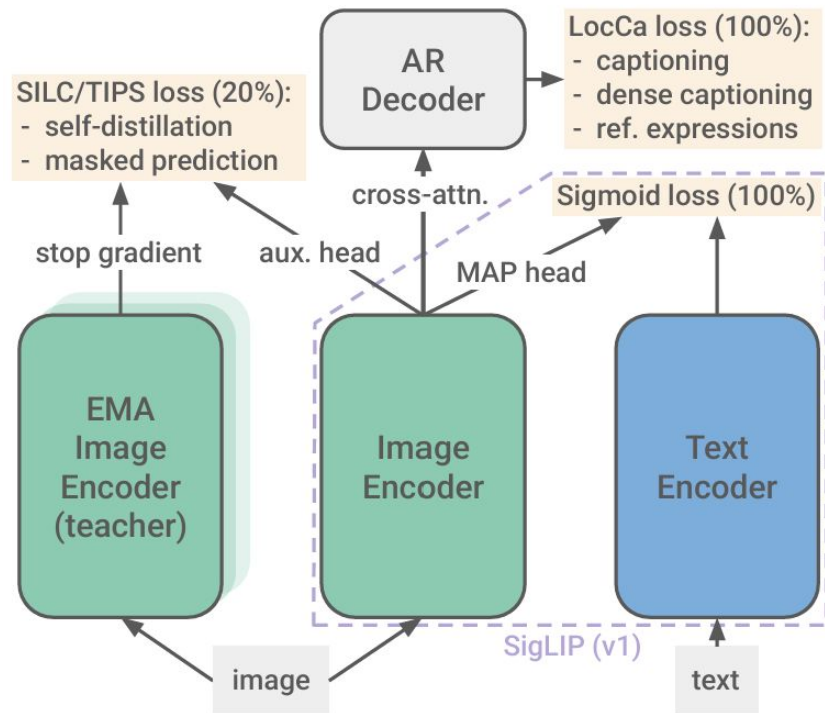


(Zhai et al. 2023)

# Architecture: Vision Encoders

## 3. Unified Recipe (**SigLIP2**)

- Combines Sigmoid, self-distillation, and captioning losses.



# Architecture: Vision Encoders

*Which ViTs are preferred for VLMs?*

1. *Cross-modal alignment is critical*
  - a. *Vision features must be compatible with language.*
  - b. *Enables seamless fusion in LLMs*
  
2. *Language-supervised encoders are preferred*
  - a. *Learn a shared vision-text embedding space*

# Architecture: Connectors

How do we connect vision encoders with LLMs?

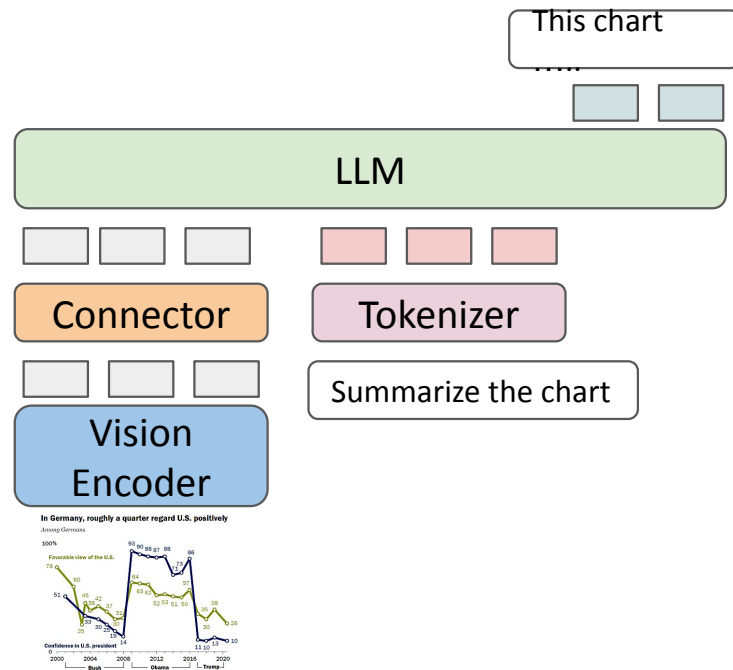
Two class of connectors:

## 1. Deep Fusion

- a. *Modifies the LLM's architecture.*
  - i. *Adds new cross-attention layers.*
- b. *Injects visual features across multiple layers*
  - i. *Enables deeper vision–language interaction*

## 2. Shallow Fusion

- a. *Projects visual features into the LLM input space*
- b. *No changes to LLM architecture*
  - i. *Simpler and more efficient*

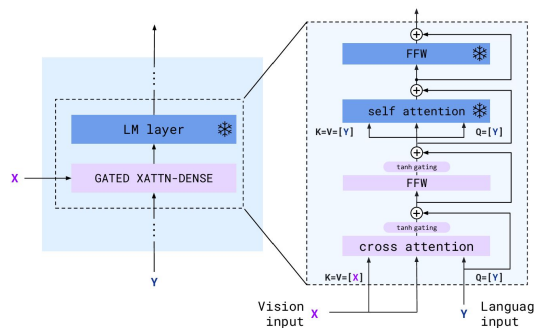


# Architecture: Connectors

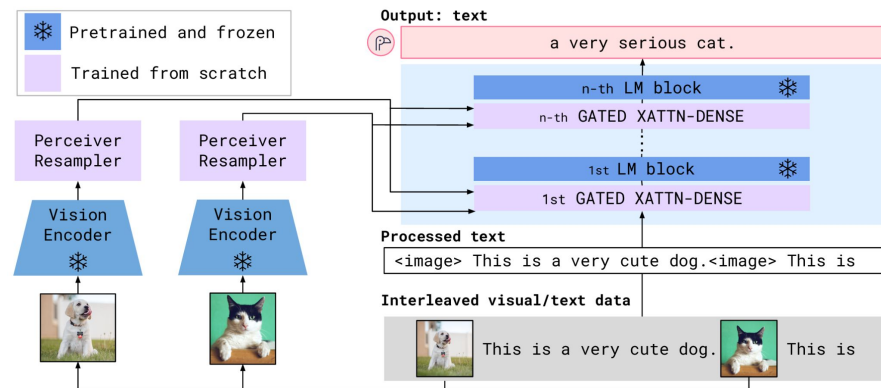
## 1. Deep Fusion

*Flamingo* (Alayrac et al. 2022)

1. *Perceiver Resampler*:
  - a. Uses learnable latent queries to attend over visual features
  - b. Produces fixed size of visual tokens.
2. *Gated attention layers*
  - a. Combines text and visual features.



GATED XATTN-DENSE (Alayrac et al. 2022)



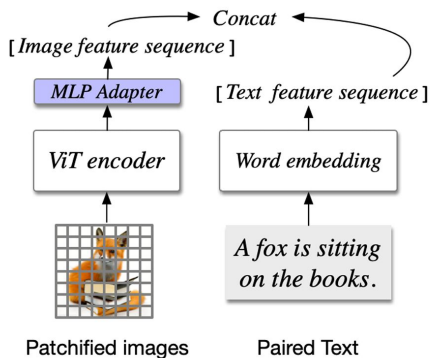
Flamingo (Alayrac et al. 2022)

# Architecture: Connectors

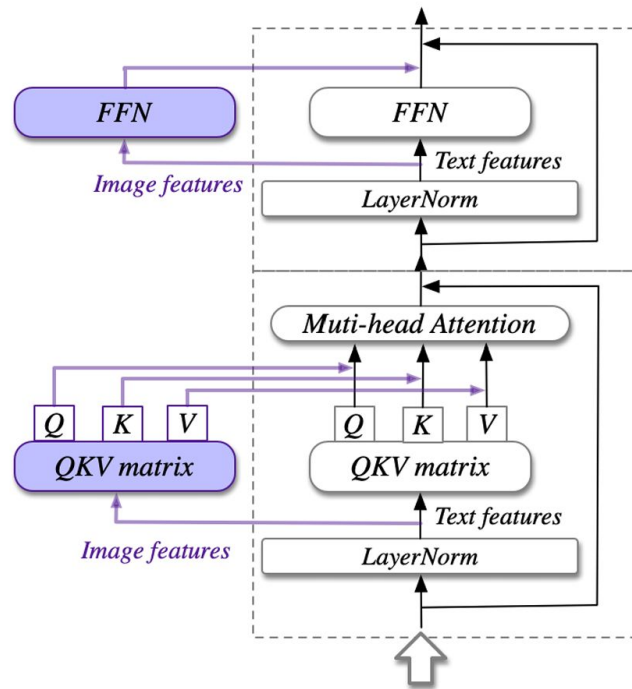
## 1. Deep Fusion

*CogVLM* (Wang et al. 2023)

1. Image is encoded by a pretrained ViT and projected into the same embedding space as text using an MLP.
2. Within the Transformer block, image features use separate QKV projections and FFN from text features



CogVLM Inputs (Wang et al. 2023)



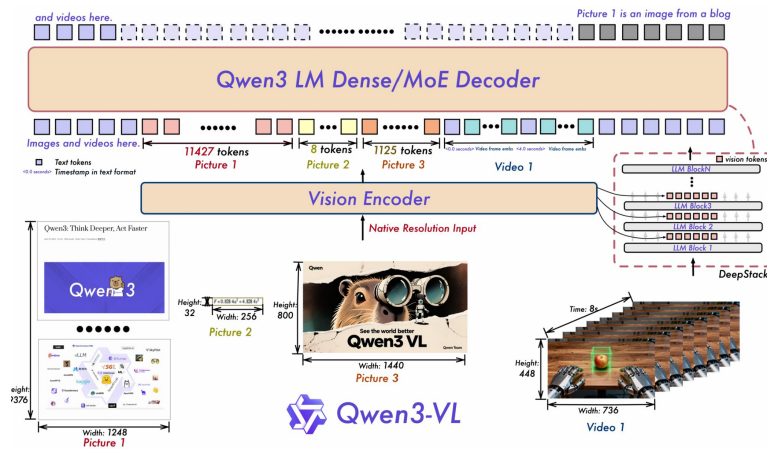
CogVLM (Wang et al. 2023)

# Architecture: Connectors

## 1. Deep Fusion

**Qwen3-VL** (Qwen Team 2025)

1. *Injects multi-level visual features into multiple LLM layers (DeepStack)*
2. *Preserves both **low** and **high-level** visual information (from intermediate ViT layers)*



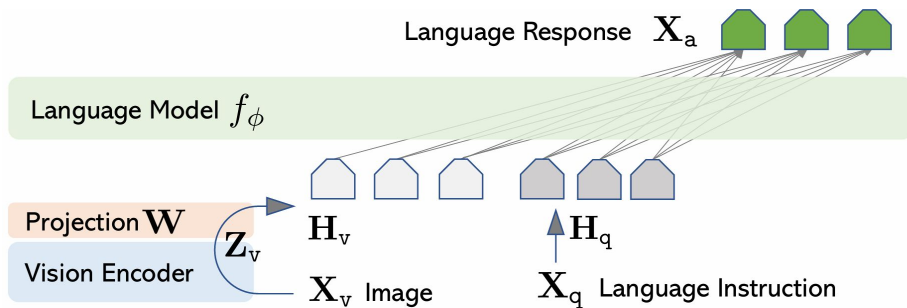
Qwen3-VL (Qwen Team 2025)

# Architecture: Connectors

## 2. Shallow Fusion

*Llava* (Liu et al. 2023)

1. Image is encoded by a pretrained ViT and projected into the same embedding space as text using either a linear layer or MLP.
2. Visual features are concatenated with text features as input to the LLM.



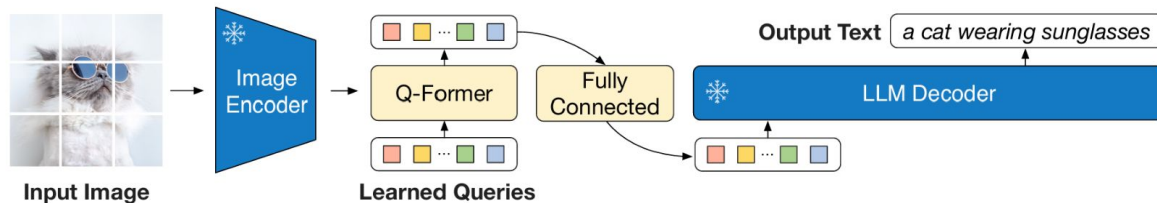
Llava (Liu et al. 2023)

# Architecture Principles: Connectors

## 2. Shallow Fusion

*Q-former* (Li et al. 2023)

1. Uses learnable query tokens to attend over image features.
2. Visual features are concatenated with text features as input to the LLM.



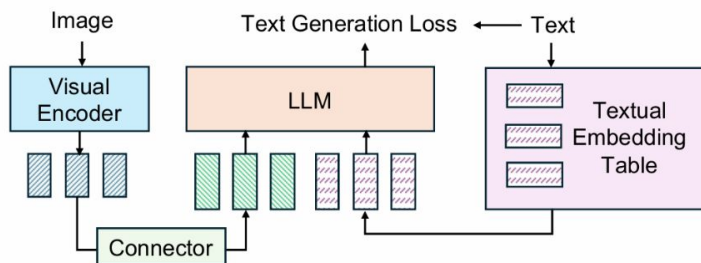
BLIP-2 (Li et al. 2023)

# Architecture: Connectors

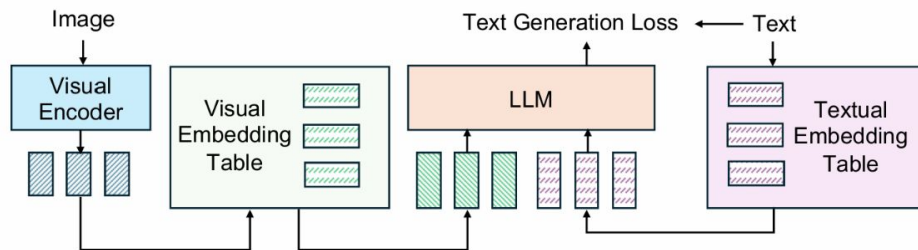
## 2. Shallow Fusion

*Ovis* (Lu et al. 2024)

1. Aligns vision and text structurally by introducing a learnable visual embedding table.
2. Mirrors how text tokens use embedding lookups



(a) Connector-based MLLM



(b) Structural Embedding Alignment in MLLM

*Ovis* (Lu et al. 2024)

Architecture: **AlignVLM Case Study**

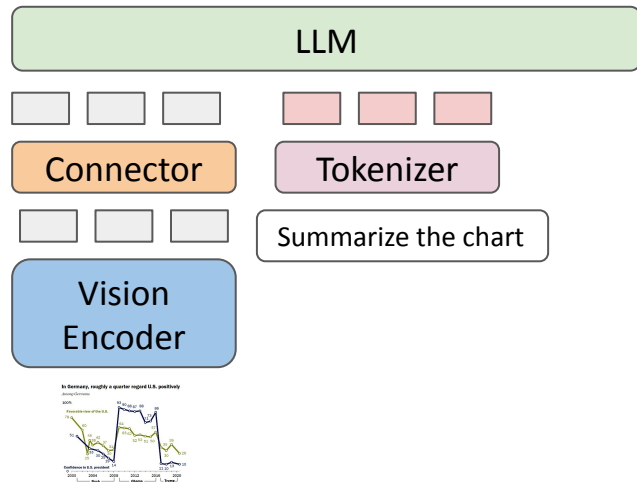
**AlignVLM: Bridging Vision and Language Latent Spaces  
for Multimodal Document Understanding**

*[NeurIPS 2025]*

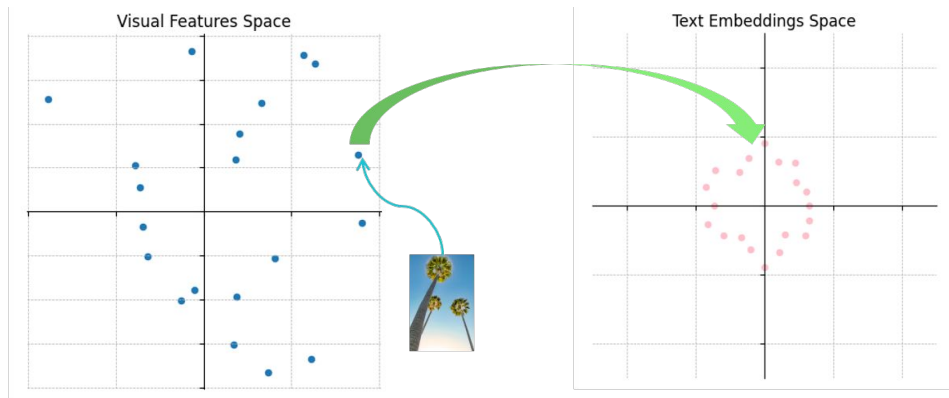
# AlignVLM Case Study: Motivation

This chart .....

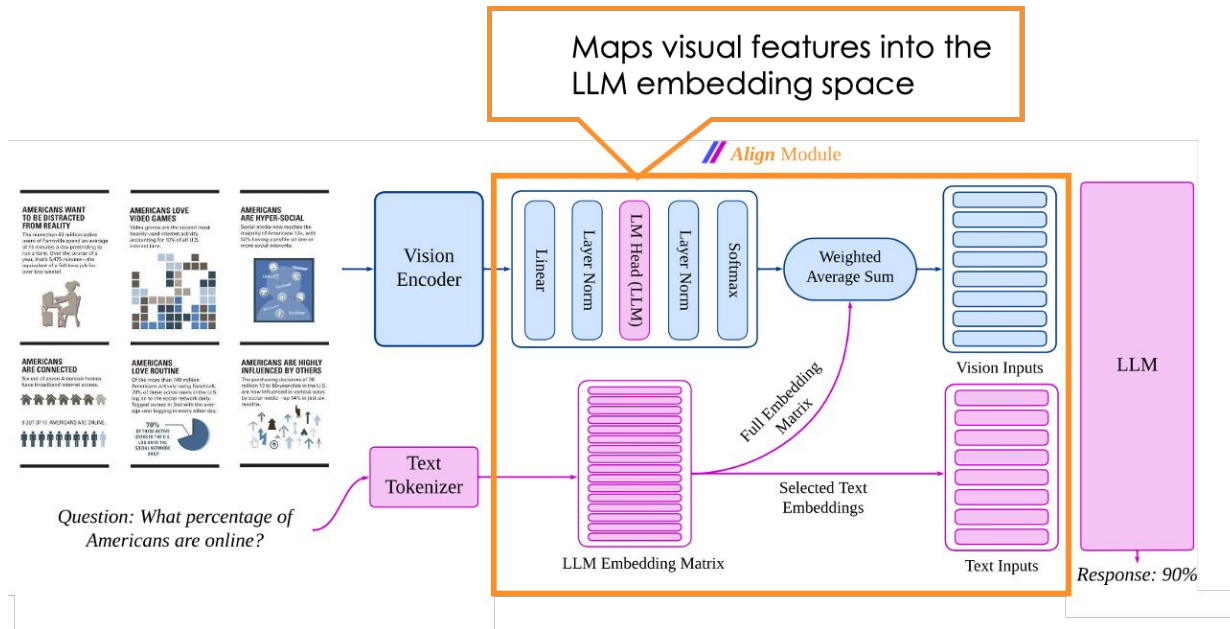
- LLM is pretrained to process a discrete set of embeddings
- Existing connector (e.g., MLP) produce continuous visual features
  - **Out-of-distribution (OOD):** making the connector data hungry!
  - **Unconstrained Mapping:** They do not enforce any hard constraints which makes them *prone to noise*.



Can we exploit the LLM's inductive bias by aligning visual features directly with text embeddings?



# AlignVLM Case Study: Solution



- Map visual features to a prob distribution over LLM token embeddings.

$$P_{\text{vocab}} = \text{softmax}(\text{LayerNorm}(W_2 \text{LayerNorm}(W_1 F))) \quad (1)$$

- Computes final features as a weighted average of text embeddings.

$$F'_{\text{align}} = P_{\text{vocab}}^\top E_{\text{text}}$$

- Constrains visual inputs to the convex hull of the LLM's embedding space, making them familiar to the LLM.

# AlignVLM Case Study: Training Setup

## Training Stages

**Stage 1:** Natural Image Understanding  
**Data:** CC-12M (Image-Caption)

**Stage 2:** Document Understanding  
**Data:** BigDocs-7.5M

**Stage 3:** Instruction Tuning for downstream tasks  
**Data:** BigDocs-Docdownstream

## Model components

- **LLM:** Llama 3.2 Family (1B, 3B, 8B)
- **Vision Encoder:** SigLip-400m

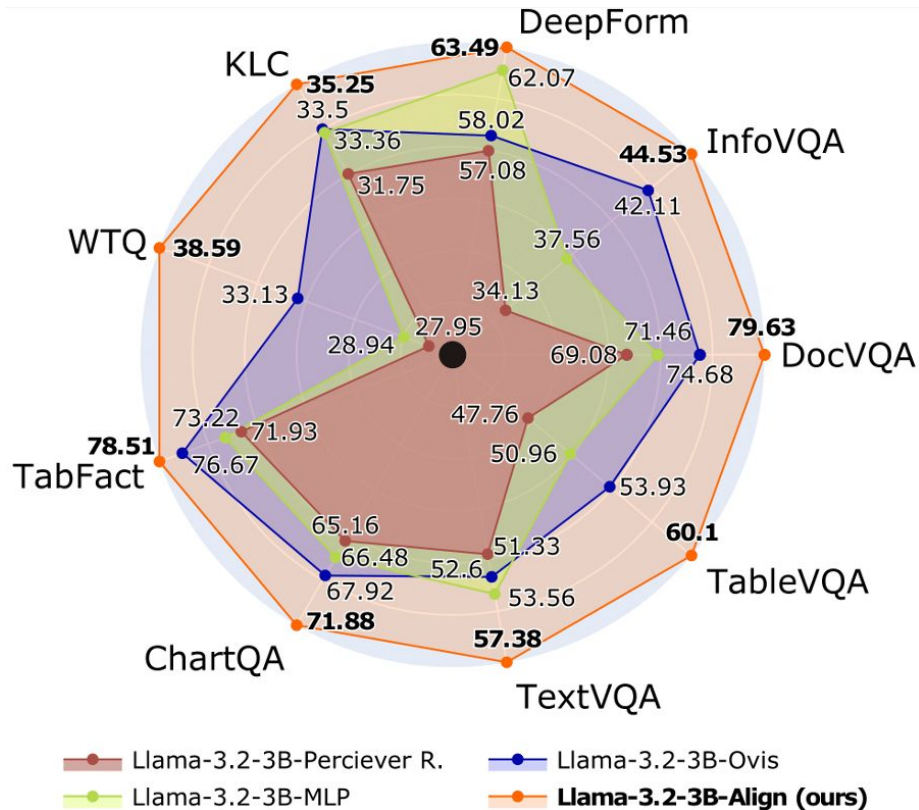
## Evaluation Benchmarks

- Nine** document benchmarks, including:
- DocVQA, InfoVQA, ChartQA, TableVQA, etc.

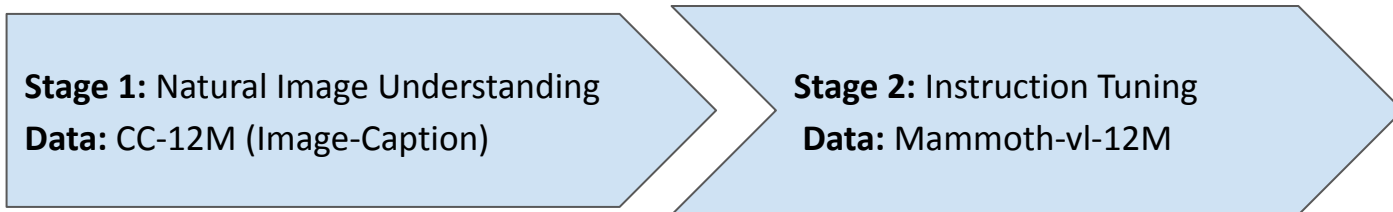
# AlignVLM Case Study: Results on Document Understanding

- We compare **our Align Module** against common connectors:
  - MLP, Perceiver Resampler, Ovis
- Trained under similar configurations to ensure a fair comparison.

The **Align Module** outperforms them all and achieves better accuracy on diverse document understanding tasks.



# AlignVLM Case Study: Results on General Vision Tasks



<b>Model</b>	<b>MMMU (dev)</b>	<b>SeedBench</b>	<b>MMVet</b>	<b>POPE</b>	<b>GQA</b>
Llama-3.2-3B-MLP	35.66	71.68	44.95	84.11	37.07
Llama-3.2-3B-ALIGN (ours)	<b>38.66</b>	<b>72.87</b>	<b>47.75</b>	<b>84.73</b>	<b>42.77</b>

# AlignVLM Case Study: Results in low-resource setup

**Stage 1:** Natural Image Understanding  
**Data:** Llava-585K (Image-Caption)

**Stage 2:** Instruction Tuning  
**Data:** Llava-Next-779K

Model	DocVQA	InfoVQA	ChartQA	TextVQA	Average	$\Delta$
LLama-3.2-3B-MLP (Llava Next)	42.11	19.93	48.44	51.97	40.61	
LLama-3.2-3B-Align (Llava Next)	71.43	30.50	69.72	65.63	59.32	+18.71
LLama-3.2-3B-MLP (BigDocs)	71.46	37.56	66.48	53.56	57.26	
LLama-3.2-3B-Align (BigDocs)	79.63	44.53	71.88	57.38	<b>63.35</b>	+6.09

# Training: Alignment and Instruction Tuning

How do we train the VLMs?

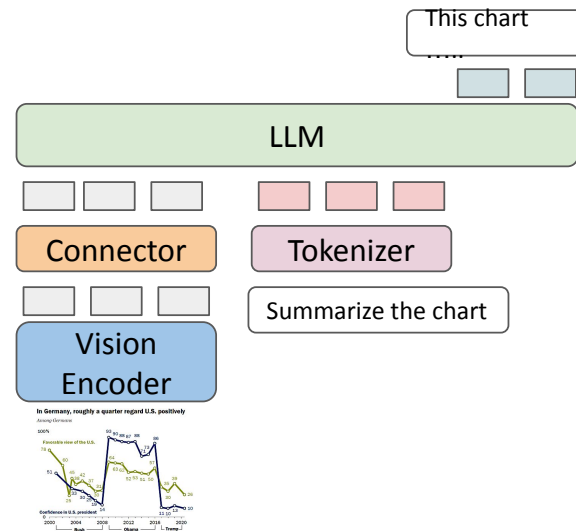
*Two-stage training process:*

## 1. Alignment:

- Image-text pairs (e.g., charts + captions).
- Freeze** vision & LLM. **Train** the connector only!

## 2. Instruction-tuning:

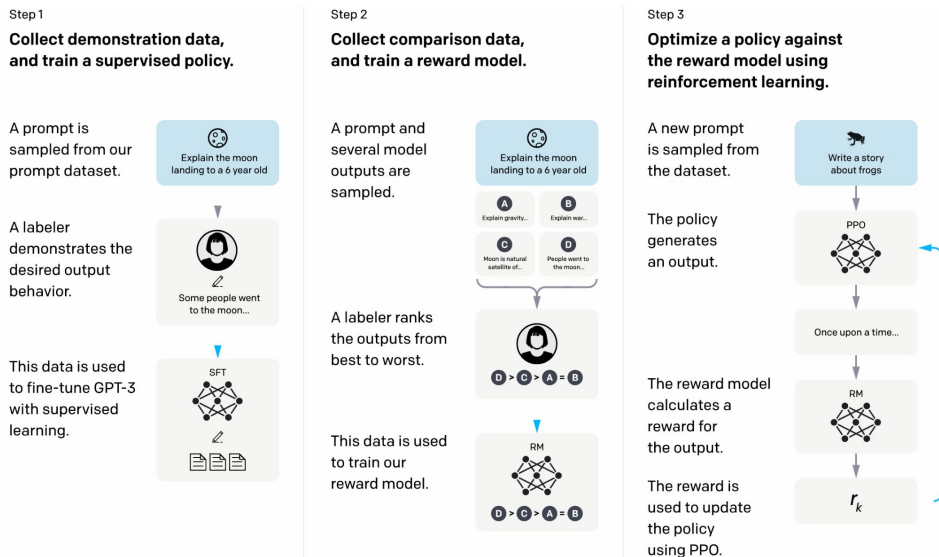
- Instruction following (e.g., QA).
- Train** LLM & Connector. **Freeze** vision.



# Training: Reinforcement Learning with Human Feedback

## Reinforcement Learning with Human Feedback (RLHF)

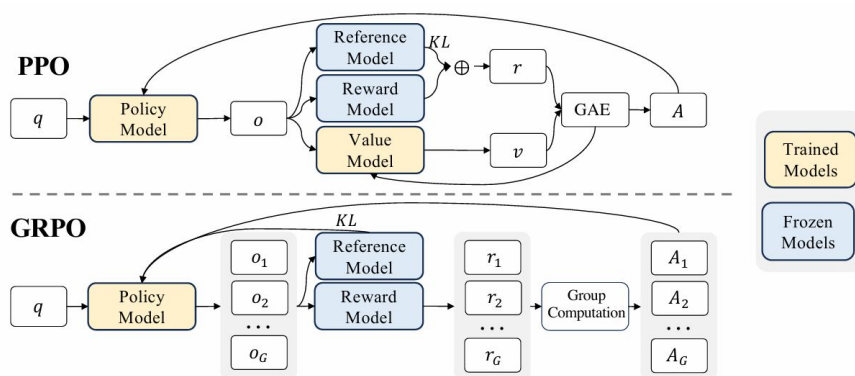
- **Goal:** Align model outputs with human preferences (e.g., removes toxicity)
- **Key Idea:** Learn a reward function from humans, then optimize the model to maximize it.



# Training: Reinforcement Learning with Verifiable Rewards

## Reinforcement Learning with Verifiable Rewards (RLVR)

- **Goal:**
  - Improve **reasoning** capabilities using **objective, programmatic rewards**
- **Key Idea:**
  - Replace human feedback with **automatically verifiable signals**
  - Reward correctness using **rules, programs, or ground truth checks**
    - Works for Math, coding, symbolic reasoning, ..etc.
    - Struggles with subjective tasks (writing, open-ended generation)



PPO vs. GRPO (Deepseek AI 2025)

# Training: How to obtain the data for training?

## Human Annotation

### Pros

- High-quality and linguistically rich annotations.
- Complex reasoning and real-world knowledge



### Cons

- Very costly.
- Scaling is challenging

## Semi-automatic Annotation

### Pros

- More scalable than human annotations.
- Preserve accuracy, nuance from human oversight



### Cons

- Diversity is limited by the generating model
- Scaling is still challenging

# Training: How to obtain data for training?

## *Synthetic Generation*

### Pros

- **Highly scalable:** LLM/VLM generation can produce millions
- **Low annotation cost:** Reduces reliance on expensive human labeling.

### Cons

- **Quality and collapse issues:** Synthetic labels may suffer from errors



Training & Data: [BigCharts-R1 Case Study](#)

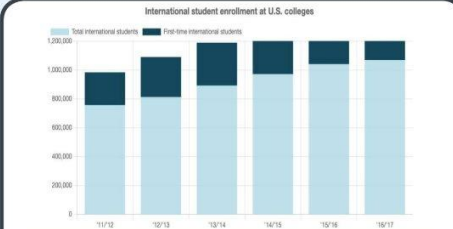
## **BigCharts-R1: Enhanced Chart Reasoning with Visual Reinforcement Finetuning**

*[COLM 2025]*

# BigCharts-R1 Case Study: Motivation

- Synthetic data for charts are generated from a single modality.
  - Chart image or underlying data table.

**A** Generated Q/A Based on Chart Image Only




International student enrollment at U.S. colleges

Legend: Total international students, First-time international students

Y-axis: 0, 200,000, 400,000, 600,000, 800,000, 1,000,000, 1,200,000

X-axis: '11/'12, '12/'13, '13/'14, '14/'15, '15/'16, '16/'17

Q: What is the total international student enrollment indicated by the **leftmost light blue bar**?

A:  **Approximately 900,000**


Captures Visual Features?

Accurate Data Values?

**B** Generated Q/A Based on Underlying Data Only

Academic Year	Total International Students	First-time International Students
'11/'12	764,495	228,467
'12/'13	820,000	280,000
'13/'14	900,000	300,000
'14/'15	980,000	350,000
'15/'16	1,050,000	370,000
'16/'17	1,078,822	290,836

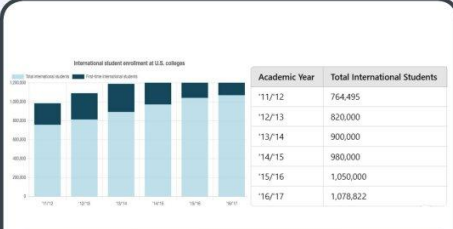
Q: What is the total international student enrollment in **11/12 academic year**?

A:  **764,495**

Captures Visual Features?

Accurate Data Values?

**C** Generated Q/A Based on Chart Image & Data (Ours)



International student enrollment at U.S. colleges


Legend: Total international students, First-time international students

Y-axis: 0, 200,000, 400,000, 600,000, 800,000, 1,000,000, 1,200,000

X-axis: '11/'12, '12/'13, '13/'14, '14/'15, '15/'16, '16/'17

Academic Year	Total International Students
'11/'12	764,495
'12/'13	820,000
'13/'14	900,000
'14/'15	980,000
'15/'16	1,050,000
'16/'17	1,078,822

Q: What is the total international student enrollment indicated by the **leftmost light blue bar**?

A:  **764,495**

Captures Visual Features?

Accurate Data Values?

Prior Works

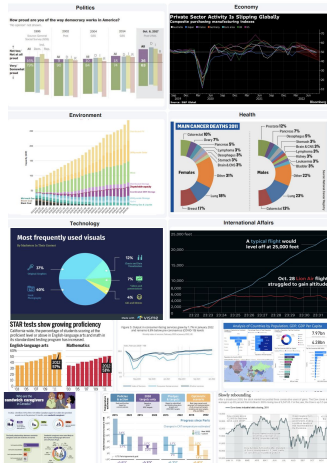
Our Approach

# BigCharts-R1 Case Study: Motivation

- Chart images that have accompanying data tables/metadata come from few sources:
  - Lack of Diversity
  - Homogenous.
- Most charts on the web do not have any associated data tables/codes
  - Visually diverse

*How can we obtain chart images that are **both** visually diverse and provide underlying metadata?*

**Real-world chart images**  
(Diverse Styles)



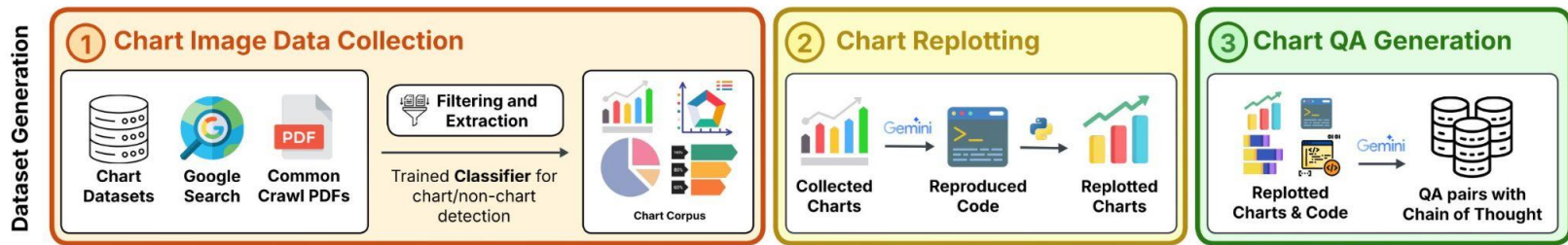
**Homogenous Chart Images**  
(Consistent Style, Real & Synthetic)



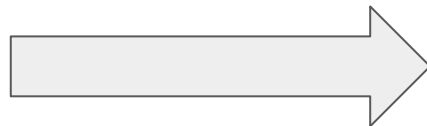
# BigCharts-R1 Case Study: Dataset Pipeline

A dataset creation pipeline that:

- Source real-world charts from multiple online platforms.
- Generates visually diverse chart images by “replotting” real-world chart images.



Original Image



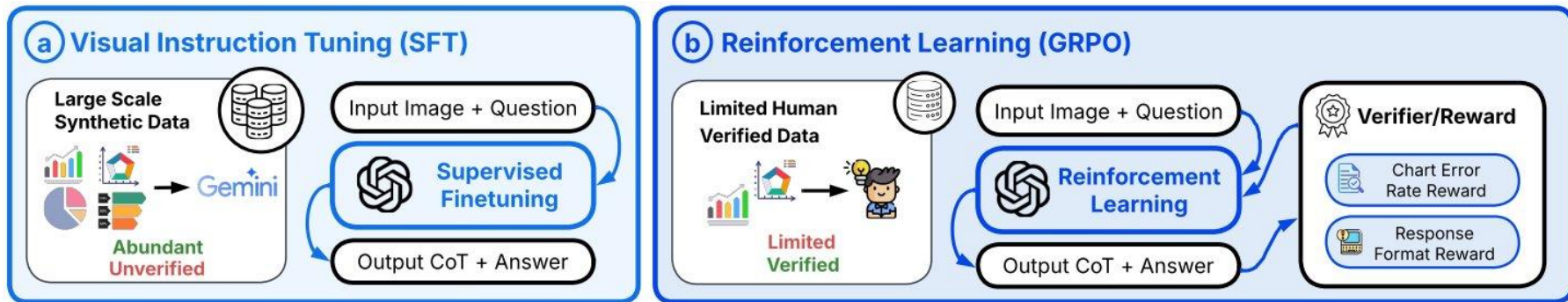
```
<code></code>
```

Code & Data



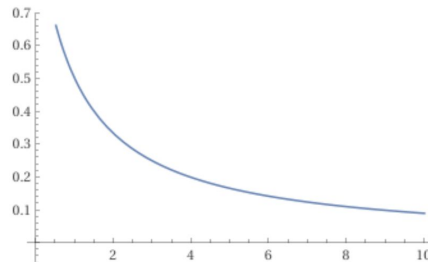
Replotted Image

# BigCharts-R1 Case Study: Training Framework



RL (GRPO) on human-labeled data to enhance chart visual math reasoning with verifiable rewards

$$ER(\hat{y}, y) = \frac{|\hat{y} - y|}{|y|}, \quad R_{\text{CERM}}(\hat{y}, y) = \begin{cases} \frac{1}{1 + ER(\hat{y}, y)}, & \text{if both } \hat{y} \text{ and } y \text{ are numeric,} \\ 1, & \text{if non-numeric and } \hat{y} = y, \\ 0, & \text{otherwise.} \end{cases}$$



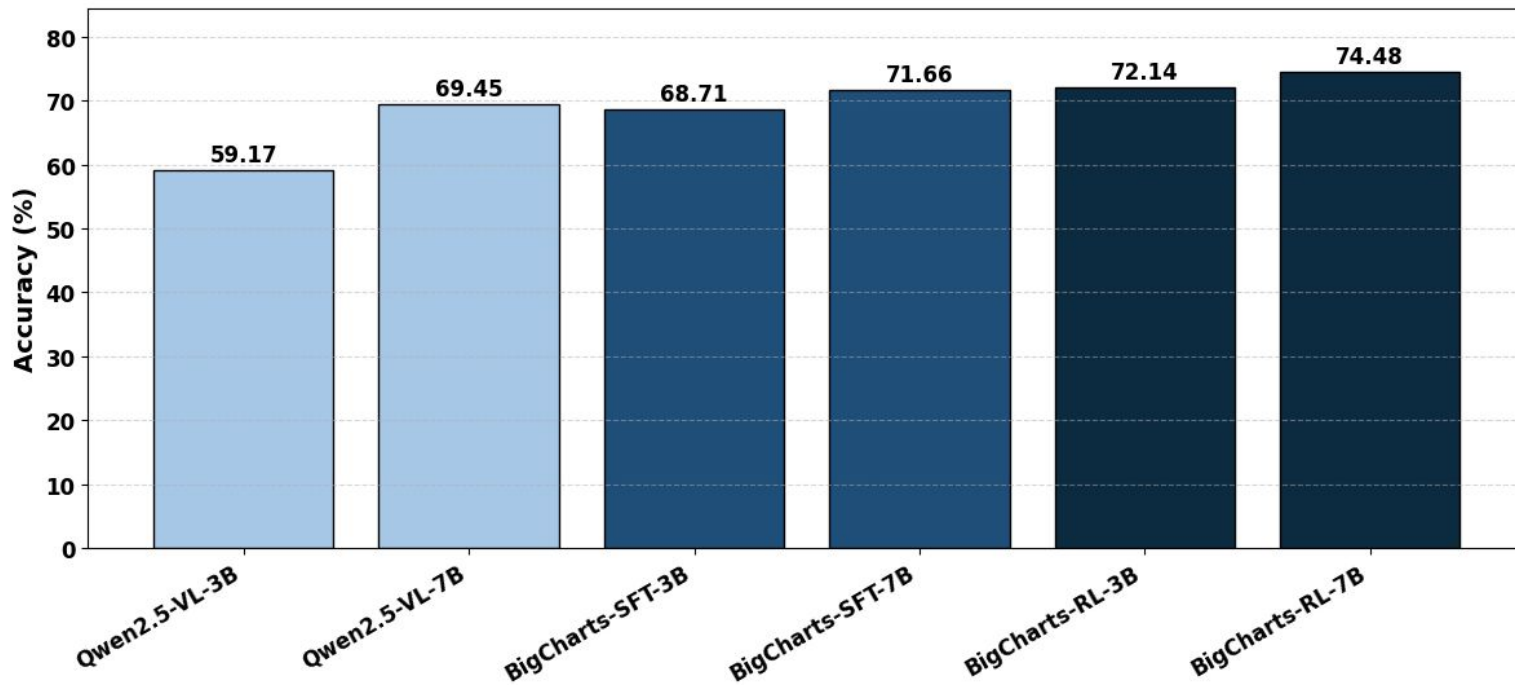
$$R_{\text{Fmt}} = \begin{cases} 1 & \text{if the model follows the required response structure,} \\ 0 & \text{otherwise.} \end{cases}$$

$$R_{\text{total}} = R_{\text{CERM}} + R_{\text{Fmt}}$$

# BigCharts-R1 Case Study: Results

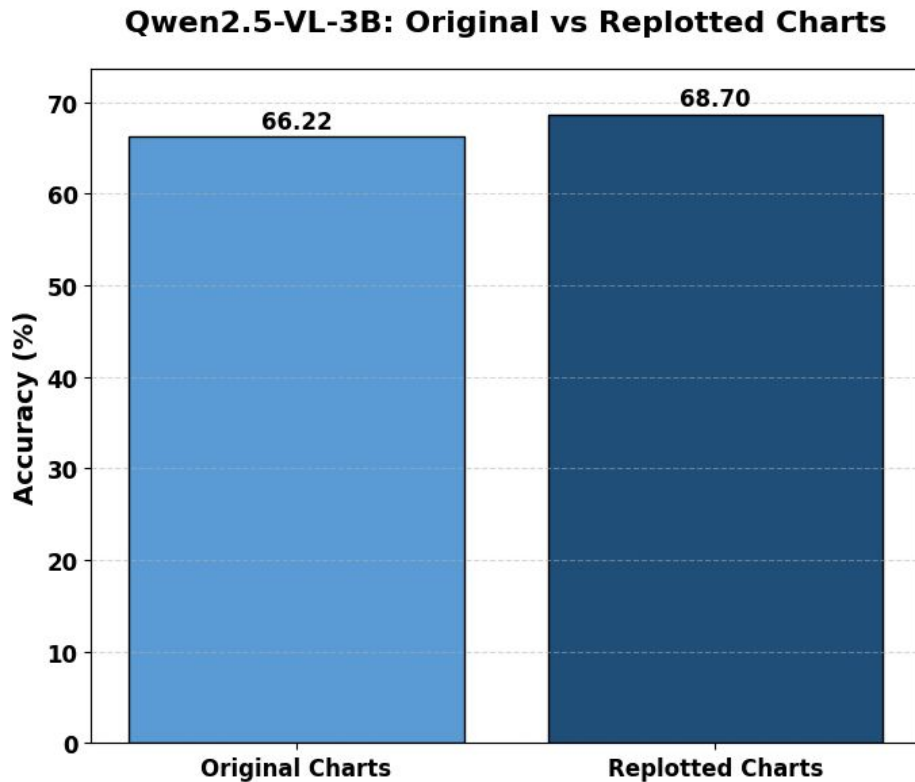
**Evaluation:** average score across five benchmarks (FigureQA, DVQA, PlotQA, ChartQA, CharXiv)

**Chart Understanding: Progress from Qwen → BigCharts-SFT → BigCharts-RL**



# BigCharts-R1 Case Study: Results

Comparison between finetuning on Q/A generated from Original Images vs Replotted Images



# Training: Can RLVR improve perception capabilities?

Can we use RLVR to improve perception as well?

## Vision-SR1

- Decomposes reasoning into visual perception + language reasoning
- Uses self-rewarding via re-prompting (perception -> reasoning)

**Question:**  
How many matryoshka's dolls are here?  
A. 14  
B. 7  
C. 12  
**Answer:** B

**Vision-R1 (GRPO CoT)**

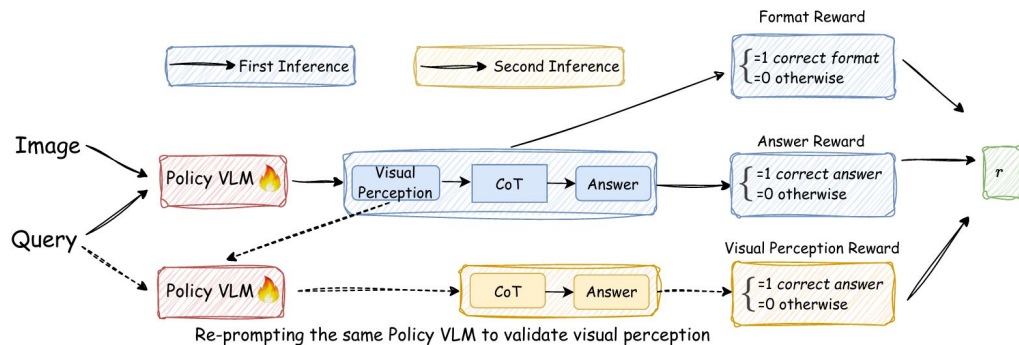

**CoT Reasoning:**  
Matryoshka dolls are nested inside each other... The leftmost large doll has 3 smaller dolls inside, and the second doll from the left has 2 smaller dolls inside, making a total of 5 dolls in the left set.  
... Reasoning generation not conditioned on visual perception.  
**MORE THINKING, MORE LANGUAGE HALLUCINATION!**

**Answer:**  
C

**Vision-SR1 (GRPO with visual perception self reward)**

**Visual Perception:**  
There are two sets of matryoshka dolls on a cardboard surface... In total, there appear to be 7 distinct matryoshka dolls (counting the outermost and innermost pieces) in the image.

**CoT Reasoning:**  
Reasoning conditioned on correct visual Perception  
Matryoshka dolls are Russian nesting dolls, so I should count each doll, including those inside the larger ones.  
...  
**Answer:**  
B



# Training: Can RLVR improve perception capabilities?

*Can we use RLVR to improve perception as well?*

## *Vision-SR1*

- *Decomposes reasoning into visual perception + language reasoning*
- *Uses self-rewarding via re-prompting (perception -> reasoning)*

Methods	General Visual Understanding					Visual Math & Hallucination			Avg.
	MMMU -Pro	MMMU	MM -Vet	RealWorld QA	VisNum Bench	Math Verse	MATH -Vision	Hallusion Bench	
Vision-SR1 (3B)	40.8	49.6	69.7	66.1	41.9	48.5	38.5	68.3	52.9
└ w/o self-reward	40.0	48.0	67.4	62.6	41.6	47.7	38.9	65.8	51.5
Vision-SR1 (7B)	49.1	57.2	76.2	71.6	42.6	56.5	46.7	69.8	58.8
└ w/o self-reward	48.8	55.3	78.4	70.9	41.4	54.8	45.3	66.4	57.7

# Tutorial Overview

## 1. FOUNDATIONS OF MLLMs

Evolution of LLMs to multimodal models; architectures, training, and alignment

## 2. MULTIMODAL REASONING

Tasks, benchmarks, and techniques for reasoning over visual documents

## 3. HUMAN-AI INTERACTION

Multimodal agents, GUI grounding, and interactive data analysis.

## 4. RESPONSIBLE & INCLUSIVE AI

Accessibility, multilingual understanding, fairness, and hallucination risks

## Future Challenges & Outlook

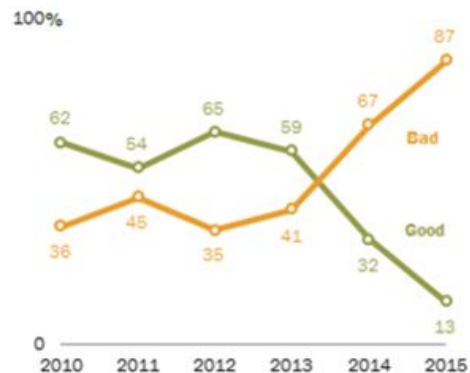
# Tasks & Benchmarks

## ChartQA (*Masry et al., 2022*)

- Real-world charts crawled from 4 online sources
- 9.6k human-authored and 23.1K Machine-generated question
  - Saturated with Clause Sonnet 3.5 achieving +90%

### Rapid Decline in Brazilians' Assessment of Economy

Current economic situation in Brazil is ...



**Question:** Which year has the most divergent opinions about Brazil's economy?

**Answer:** 2015

**Question:** What is the peak of the orange line?

**Answer:** 87

# Tasks & Benchmarks

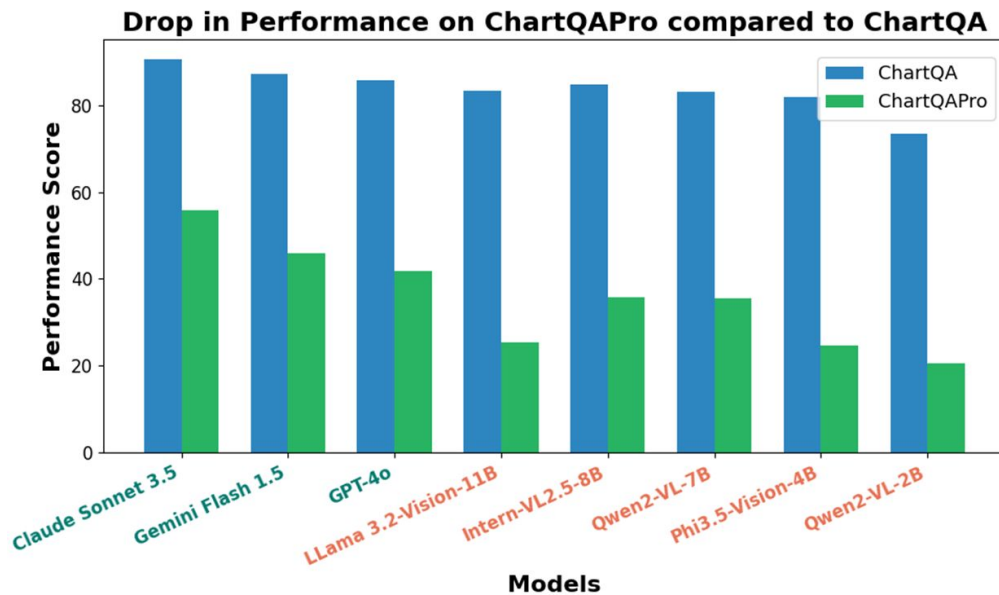
## ChartQAPro (Masry et al. 2025):

- 1,341 charts and from 99 diverse sources
- More diverse chart types including infographics and dashboards
- 1,948 questions with 8+ question types



# Tasks & Benchmarks

- *Drop in Performance on ChartQAPro*
- *Chart Question Answering is far from solved!*



# Tasks & Benchmarks

CharXiv (Wang et al. 2024)

- Scientific Charts Question Answering with 2.3K QA pairs

Example

Question: For the subplot at row 1 and column 1, what are the names of the labels in the legend?

- You should write down the labels from top to bottom, then from left to right and separate the labels with commas. Your final answer should account for only labels relevant to the plot in the legend, even if the legend is located outside the plot.
- If the plot does not have a legend or no legend is not considered relevant to this plot, answer "Not Applicable".

Answer: Not Applicable

Example

Question: For the subplot at row 5 and column 2, what is the difference between the maximum and minimum values of the tick labels on the continuous legend (i.e., colorbar)?

- You should remove the percentage sign (if any) in your answer.
- If the plot does not have an explicit colorbar-based continuous legend or the legend is not considered relevant to this subplot, answer "Not Applicable".

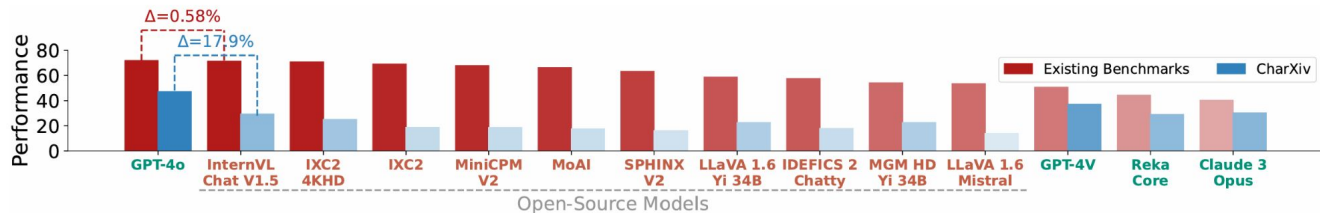
Answer: 0.8

Example

Question: For the bottom right subplot, what is the difference between the maximum and minimum values of the tick labels on the continuous legend (i.e., colorbar)?

- You should remove the percentage sign (if any) in your answer.
- If the plot does not have an explicit colorbar-based continuous legend or the legend is not considered relevant to this subplot, answer "Not Applicable".

Answer: Not Applicable



# Tasks & Benchmarks: Open-ended Question Answering

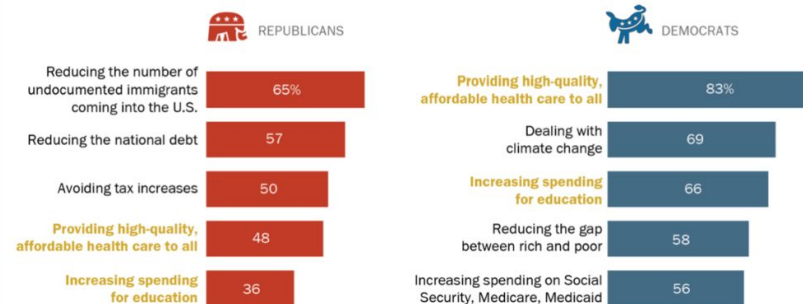
OpenCQA (*Kantharaj et al., 2022*)

- 7,724 human-written questions about charts and the associated descriptive answers

Question: Compare the Democrats and Republicans views about providing health care to the population?

**Republicans and Democrats have different ideas about what government should do to improve the lives of future generations of Americans**

% of **Republicans/Democrats** saying each of the following should be a **top priority** in order for the federal government to improve the quality of life for future generations



## Answer

While 83% of Democrats say providing high-quality, affordable health care for all should be a top priority, a much smaller share of Republicans (48%) agree.



# Tasks & Benchmarks: Infographic Question Answering

InfoVQA (Mathew et al., 2021)

- 30K QA on 5.4K real-world infographics.



How many companies have more than 10K delivery workers?

Answer: 2

Evidence: [Figure](#)

Answer-source: [Non-extractive](#) Operation: [Counting](#) [Sorting](#)

Who has better coverage in Toronto - Canada post or Amazon?

Answer: canada post

Evidence: [Text](#)

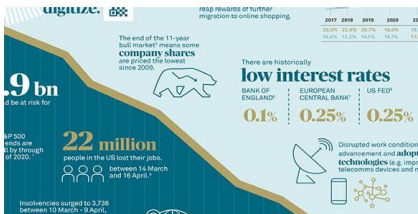
Answer-source: [Question-span](#) [Image-span](#) Operation: [none](#)

In which cities did Canada Post get maximum media coverage?

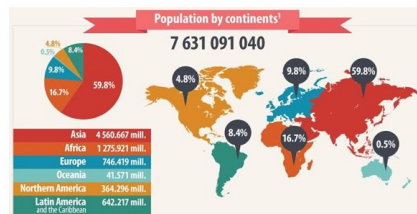
Answer: vancouver, montreal

Evidence: [Text](#) [Map](#)

Answer-source: [Multi-span](#) Operation: [none](#)



What is the interest rates of European Central Bank and US FED?



Which is the least populated continent in the world?

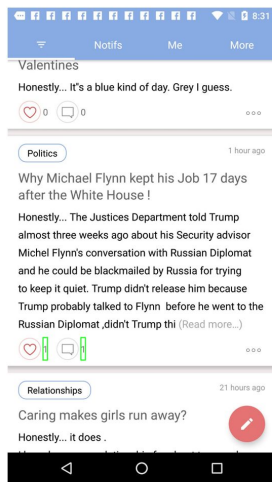


What percentage of workers are not working from home?

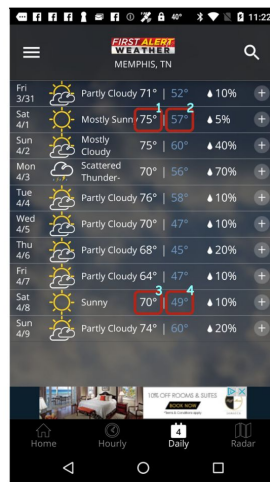
# Tasks & Benchmarks: Mobile Screen Question Answering

ScreenQA (Hsiao et al., 2022)

- 86K QA questions on mobile-app screenshots!



(a) Question: "How many likes and comments are there for the post *Why Michael Flynn ...?*"



(b) Question with Ambiguity: "What's the temperature on Saturday?"

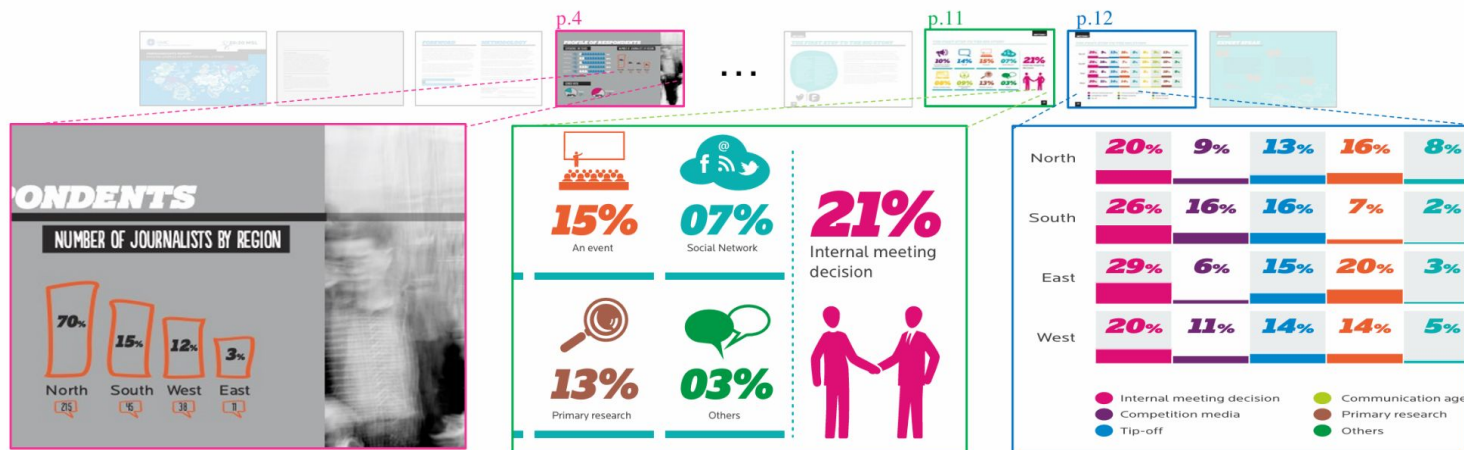


(c) No answer to the question: "What is the date of version 1.3.1?"

# Tasks & Benchmarks: Slide Question Answering

SlideVQA (*Tanaka et al., 2023*)

- 14.5K questions over 54K slides!



Q: What is the **tip-off media percentage** in the **region with 70% of journalists** and **South**?

A: 13%, 16%

Evidence pages: 4, 12

Answer type: Multi-Span Reasoning type: Multi-hop

Q: What is the **percentage of the internal meeting decision**?

A: 21%

Evidence pages: 11

Answer type: Single-Span Reasoning type: Sing-hop

Q: What is the difference in the **competition media percent age** between **East** and the **region with 12% of journalists**?

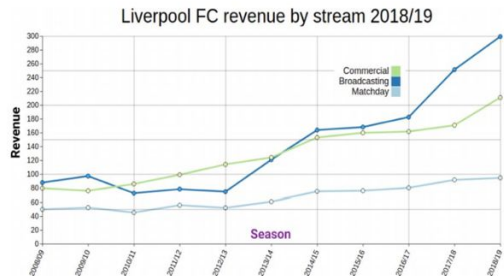
A: 5% (11% - 6%)

Evidence pages: 4, 12

Answer type: Non-Span Reasoning type: Multi-hop, Numerical

# Tasks & Benchmarks: Text Generation from Visuals

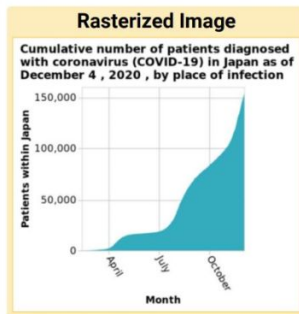
## Chart Summarization



Broadcasting is the largest source of revenue for Liverpool FC. In 2018/2019, the club earned approximately 299.3 million euros

### Chart-to-text

[Kantharaj et al, ACL 2022]



### Data Table

Cumulative number of patients diagnosed with coronavirus (COVID-19) in Japan as of December 4, 2020, by place of infection

Month	Patients within Japan
Feb 11, 2020	16
...	...

### Scene Graph

```
{title: "Cumulative number ...", x: -76, y: -50,},  
{x-axis: "Month", x: 100, y: 55.6,},  
{y-axis: "Patients within Japan", x: ...},  
{x-tick: [{"x: 33, val: "April"}, ...]},  
marks: [...],  
...}
```

### Generated L1 Caption

Here is an area chart is labeled Cumulative number of patients diagnosed with coronavirus (COVID-19) in Japan as of December 4, 2020, by place of infection. On the x-axis, Month is measured with a categorical scale starting with April and ending with October. There is a linear scale with a minimum of 0 and a maximum of 150,000 along the y-axis, labeled Patients within Japan.

### Crowdsourced L2/L3 Caption

By December 4th 2020, approximately 160,000 people in Japan had been diagnosed with COVID-19. The first person diagnosed with COVID-19 in Japan was diagnosed in March 2020. The greatest increase in cumulative number of patients in Japan diagnosed with COVID-19 occurred between November and December 2020.

## VisText

[Tang et al, ACL 2023]

# Tasks & Benchmarks: Text Generation from Visuals

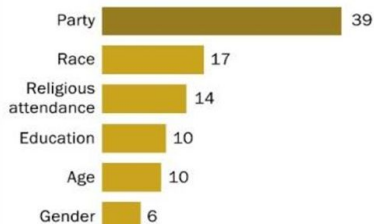
**Chart-to-text:** 44.1K Chart-summary pairs from Statista and Pew Research Center

## Problem Setup #1

### Chart:

#### Partisan gaps dwarf race, education, other differences in political values

Average percentage point gap across 30 political values items by ...



Notes: Indicates average gap between the share of two groups taking the same position across 30 values items. Party=difference between Rep/Lean Rep and Dem/Lean Dem. Race=white non-Hispanic vs. black non-Hispanic. Religious attendance=attend religious services weekly or more vs. attend less often. Education=college grad vs. non-college grad. Age=18-49 vs. 50+. Source: Survey of U.S. adults conducted Sept 3-15, 2019.

PEW RESEARCH CENTER

### Table:

Demographic	Average Percentage Point Gap
Party	39
Race	17
Religious Attendance	14
Education	10
Age	10
Gender	6

### Metadata:

- Title: Partisan gaps dwarf race, education, other differences in political values
- Chart type: Bar
- Topic: U.S. Politics & Policy

### Gold Summary:

Across all 30 political values, the differences between Republicans and Democrats dwarf all other differences by demographics or other factors. The 39-point average gap is more than twice the difference between white and nonwhite adults (17 percentage points); people who regularly attend religious services and those who do not (14 points); college graduates and those who have not completed college (10 points); younger and older adults (also 10 points); and men and women (6 points).

# Tasks & Benchmarks: Text Generation from Visuals

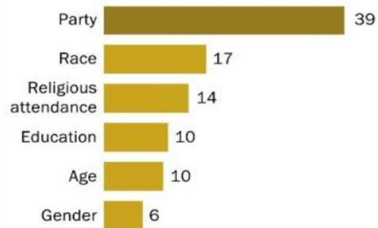
**Chart-to-text:** 44.1K Chart-summary pairs from Statista and Pew Research Center

## Problem Setup #2

### Chart:

#### Partisan gaps dwarf race, education, other differences in political values

Average percentage point gap across 30 political values items by ...



Notes: Indicates average gap between the share of two groups taking the same position across 30 values items. Party=difference between Rep/Lean Rep and Dem/Lean Dem. Race=white non-Hispanic vs. black non-Hispanic. Religious attendance=attend religious services weekly or more vs. attend less often. Education=college grad vs. non-college grad. Age=18-49 vs. 50+. Source: Survey of U.S. adults conducted Sept 3-15, 2019.

PEW RESEARCH CENTER

### Table:

Demographic	Average Percentage Point Gap
Party	39
Race	17
Religious Attendance	14
Education	10
Age	10
Gender	6

### Metadata:

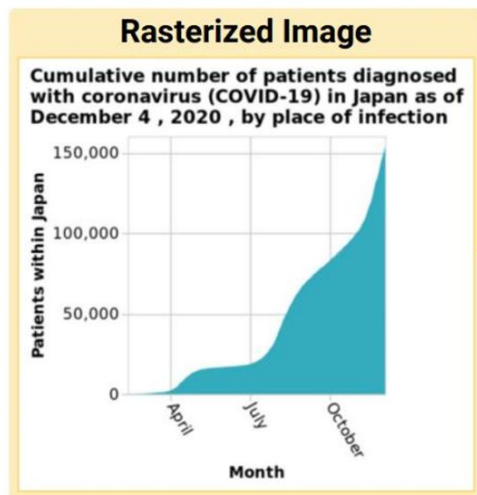
- Title: Partisan gaps dwarf race, education, other differences in political values
- Chart type: Bar
- Topic: U.S. Politics & Policy

### Gold Summary:

Across all 30 political values, the differences between Republicans and Democrats dwarf all other differences by demographics or other factors. The 39-point average gap is more than twice the difference between white and nonwhite adults (17 percentage points); people who regularly attend religious services and those who do not (14 points); college graduates and those who have not completed college (10 points); younger and older adults (also 10 points); and men and women (6 points).

# Tasks & Benchmarks: Text Generation from Visuals

- **VisText:** 12.4K Charts with generated+crowdsourced caption
  - *Scene graph a hierarchical representation of a chart's visual elements*



**Data Table**

Cumulative number of patients diagnosed with coronavirus (COVID-19) in Japan as of December 4, 2020, by place of infection

Month	Patients within Japan
Feb 11, 2020	16
...	...

**Scene Graph**

```
{title: "Cumulative number ...", x: -76, y: -50,},  
axes: [{x-axis: "Month", x: 100, y: 55.6,},  
       {y-axis: "Patients within Japan", x: ...}  
       {x-tick: [{x: 33, val: "April"}, ...]},  
marks: [...],  
...}
```

**Generated L1 Caption**

Here is a area chart is labeled Cumulative number of patients diagnosed with coronavirus (COVID-19) in Japan as of December 4, 2020, by place of infection. On the x-axis, Month is measured with a categorical scale starting with April and ending with October. There is a linear scale with a minimum of 0 and a maximum of 150,000 along the y-axis, labeled Patients within Japan.

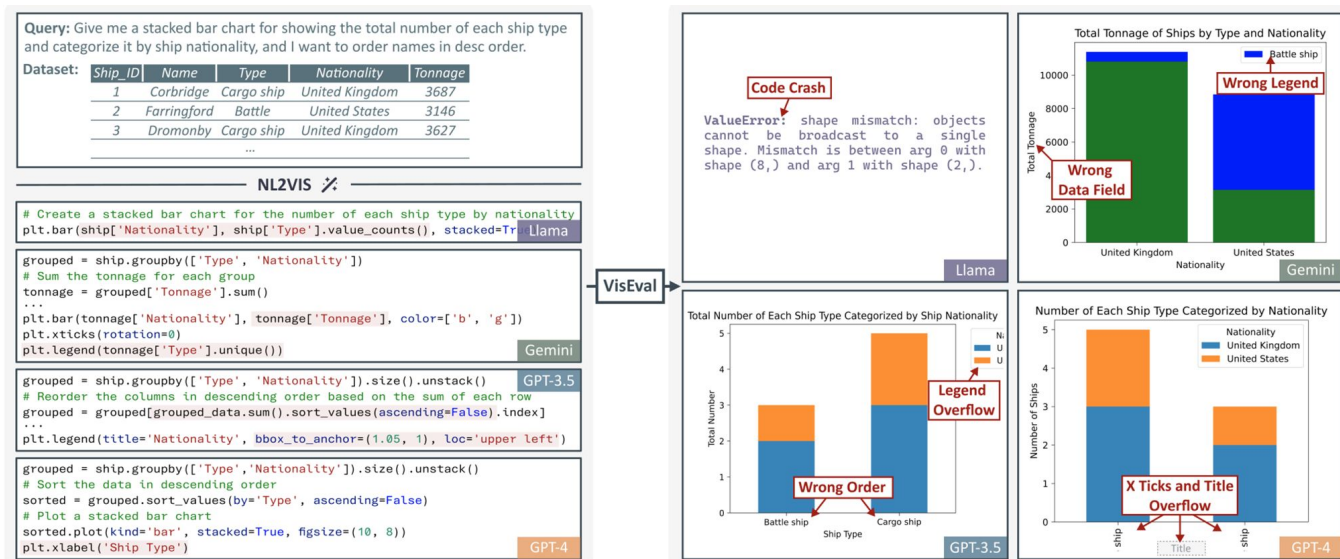
**Crowdsourced L2/L3 Caption**

By December 4th 2020, approximately 160,000 people in Japan had been diagnosed with COVID-19. The first person diagnosed with COVID-19 in Japan was diagnosed in March 2020. The greatest increase in cumulative number of patients in Japan diagnosed with COVID-19 occurred between November and December 2020.

# Tasks & Benchmarks: Visualization Generation

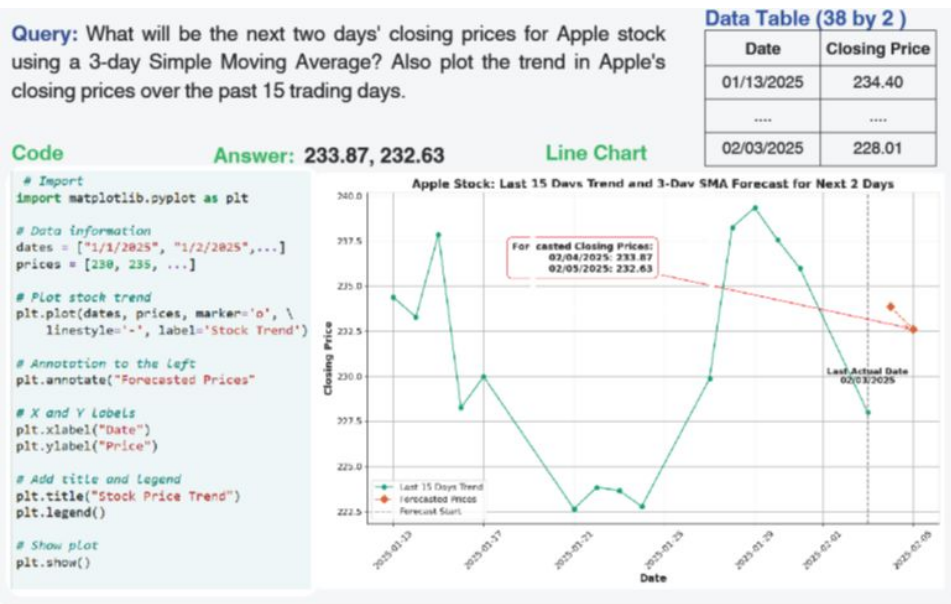
NL2Vis (Chen et al. 2024):

- 2,524 queries covering seven chart types.



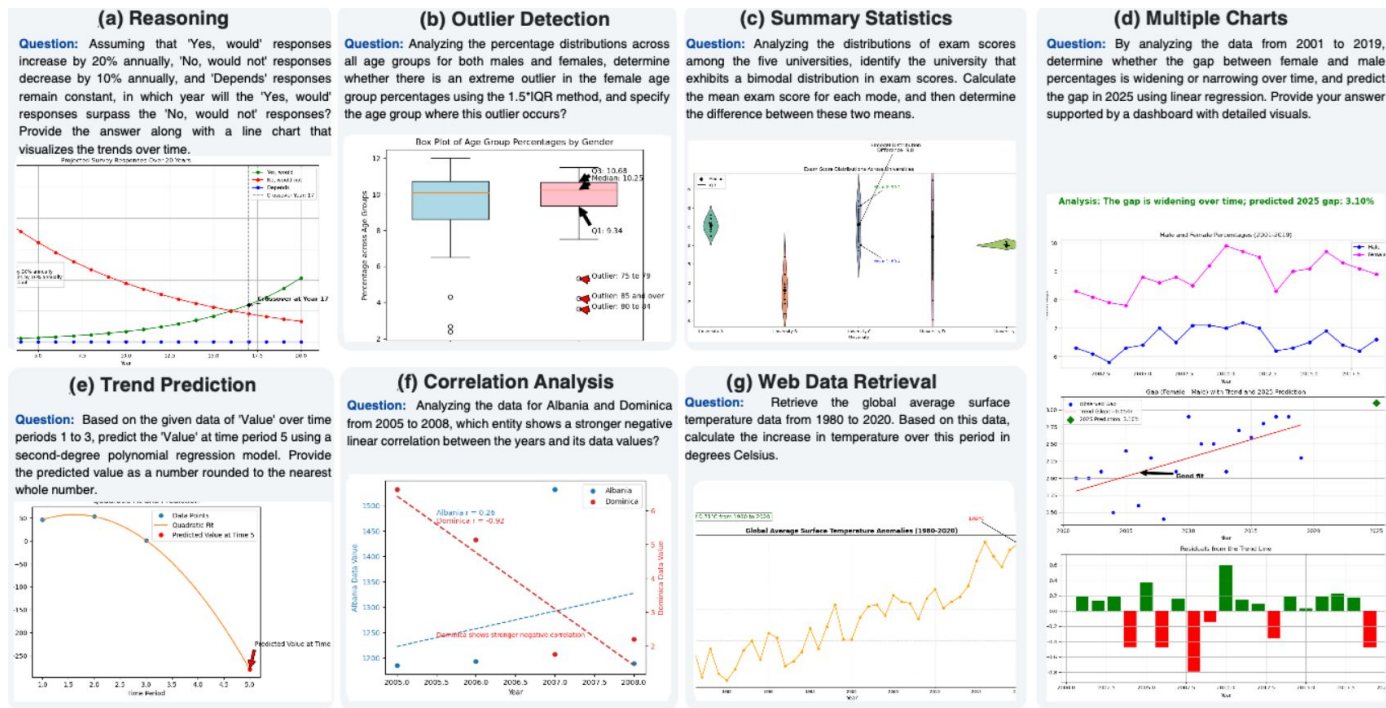
# Tasks & Benchmarks: Visualization Generation

- Text2Vis ([Rahman et al. 2025](#))
  - Exploratory analysis in Data Science workflow.
  - 1985 queries covering 20+ chart types.



# Tasks & Benchmarks: Visualization Generation

- Text2Vis (Rahman et al. 2025)



# Tasks & Benchmarks: Visualization Generation

- Text2Vis (Rahman et al. 2025)
  - Multi-agent framework for error analysis and feedback

### 1. Initial Response Generation

**Data Table... .. User Input**

- Characteristic, Annual turnover in million euros... ..
- ... Years 2008-2016
- .... Turnover Values 191985.5, 143670.0, 178858.4

**Query:** From the annual turnover data from 2008 to 2016, ... determine the year in which the annual turnover deviated the most from this average, and compute the value of that deviation in million euros.

**Generating Response... ..**


1. Answer: 5841

2. Code: 

```
import matplotlib.pyplot as plt
years = ['2008', '2009', '2010' ... ..]
turnovers = [191985.5, 143670.0, .....]
plt.figure(figsize=(10, 6))
plt.plot(years, turnovers, marker='o', .....)
```

....  
Plt.show()

*Missing Numpy Import,  
Incorrect Annotation*




Actor

### 2. Error Analysis & Feedback

**Response Validation... ..**

1. **Answer Feedback:** Incorrect calculation. The ... computed as  $\text{Max}(\text{abs}(\text{Turnover} - \text{Average Turnover}))$ .  
Mistake Type: Calculation Error  
The maximum deviation year should be recomputed .....

2. **Code Feedback:** Missing import : Import numpy as np ...  
Incorrect Annotation Formatting: 'Max Deviation: 2009  
....[max\_deviation\_index]:.1f} million euros'  
Should be formatted in a single line or properly broken.



Critic


### 3. Revised Response Generation

**Correcting Response... ..**


Answer: 2009, 50023.76 million euros

Code: 

```
import numpy as np
import matplotlib.pyplot as plt
years = ['2008', '2009', '2010' ... ..]
deviations=[abs(turnover-average_turnover)
m...]
max_index=deviatis.index(max(deviations))
plt.plot(years,turnovers, marker='o',....)
plt.annotate(f'MaxYear
{deviations[max_index]:.1f}
```



Actor



Annual Turnovers from 2008 to 2016

Final Answer:  
2009, 50023.8

# Tasks & Benchmarks: Visualization Generation

- Text2Vis (Rahman et al. 2025)

Model Setup	Strategy	Code Exec. Success (%)	Answer Match (%)	Clarity Readability	Chart Correctness	Final Pass Rate (%)
<i>(A) Baseline</i>						
GPT-4o	0-shot	87	42	3.45	3.15	26
Gemini 1.5 Flash	0-shot	83	34	3.30	2.90	17
GPT-4o	3-shot	88	42	3.45	3.15	26
Gemini 1.5 Flash	3-shot	81	29	3.36	3.38	20
GPT-4o	RAG + 3-shot	88	38	3.65	3.75	31
Gemini 1.5 Flash	RAG + 3-shot	80	31	3.30	3.45	22
<i>(B) Agentic Inference (LLM Feedback)</i>						
GPT-4o + Gemini 1.5	Answer + Code	91	49	3.85	3.87	36
<b>GPT-4o + GPT-4o</b>	Answer + Code	<b>94</b>	<b>53</b>	<b>3.99</b>	<b>4.02</b>	<b>42</b>
GPT-4o + GPT-4o	Answer + Code + Visual	93	46	4.02	4.23	41
<i>(C) LLM Feedback Ablation</i>						
GPT-4o + GPT-4o	Answer Only	86	47	3.51	3.20	28
GPT-4o + Matplotlib	Code Exec Only	94	37	3.96	4.02	34
GPT-4o + GPT-4o	Code Only	94	36	3.99	4.19	32
GPT-4o + GPT-4o	Visual Only	94	38	4.03	4.24	33

# Reasoning Techniques

- Chain of Thought (Wei et al. 2022)
- Program of Thought (Chen et al. 2022)

INPUT: VISUAL CONTEXT

Share of people who say university is more important for boys  
*Percentage agreeing with "University is more important for a boy than a girl"*

Country	Percentage
Malaysia	43%
Philippines	38.92%
Ghana	27.58%
Switzerland	8.82%

INPUT: PROMPT

User: "What is the average share of people in Philippines and Ghana who think University is more important for boys?"

**Direct Generation**

The model produces the answer immediately in a single step, without revealing its internal process.

MODEL OUTPUT

"The average is 33.25%."

**Hidden Logic:**  $(38.92 + 27.58) / 2 = 33.25$   
Process is opaque to the user.

CHARACTERISTICS

- ⊙ Fast & Efficient
- ⊗ Less effective for reasoning
- ⊗ Hard to verify

**Chain-of-Thought (CoT)**

The model articulates intermediate reasoning steps in natural language before giving the final answer.

MODEL OUTPUT

1. Philippines share is 38.92%.
2. Ghana share is 27.58%.
3. Sum:  $38.92 + 27.58 = 66.50$ .
4. Average:  $66.50 / 2 = 33.25$ .

"The average is 33.25%."

CHARACTERISTICS

- ⊙ Human-interpretable
- ⊙ Handles multi-step logic
- ⊗ Still prone to arithmetic errors

**Program-of-Thought (PoT)**

The model generates executable code (Python/SQL) to perform calculations deterministically.

MODEL CODE OUTPUT

```
phi_val = 38.92
gha_val = 27.58
avg = (phi_val + gha_val) / 2
print(f"{avg:.2f}%")
```

>> 33.25%

CHARACTERISTICS

- ⊙ High arithmetic precision
- ⊙ Resolves ambiguity
- ⊙ Requires execution environment

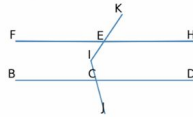
# Reasoning Techniques

- **Visual Chain of Thought** (Hu et al. 2024)
  - Mimics how human draw to find patterns.

## Geometry



Given  $\angle BCJ = 105^\circ$ ,  
 $\angle KEF = 105^\circ$ . Find  $\angle EIC$

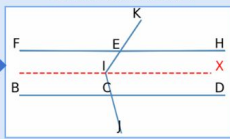


**THOUGHT 1:** I need to draw an auxiliary Line IX parallel to Line BD

**ACTION 1:** 🛠️ Draw the aux. line using Python Code

```
# Define auxiliary point X for the line IX
x_I = points["I"][0]
y_I = points["I"][1]
x_X = x_I + 100
y_X = y_I + slope_bd * (x_X - x_I)
# Plot the auxiliary line IX in red color
ax.plot([x_I, x_X], [y_I, y_X], 'red') ...
```

### OBSERVATION 1



**THOUGHT 2:** Given  $\angle BCJ = \angle KEF = 105^\circ$ , their supplementary angles,  $\angle KEH = \angle DCJ = 180^\circ - 105^\circ = 75^\circ$ . Given BD parallel to IX, it follows that  $\angle EIX = \angle KEH$ , and  $\angle XIC = \angle DCJ$ , both measuring  $75^\circ$ . Therefore,  $\angle EIC = \angle EIX + \angle XIC = 150^\circ$

## Graph Connectivity



Is there a path between node 0 and node 4 in the graph?

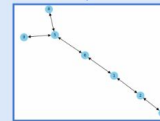
adjacency matrix =  $\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$

**THOUGHT 1:** We can use the networkx library to plot the graph

**ACTION 1:** 🛠️ Call Python Plotting tool

```
G = nx.from_numpy_matrix(np.array(adjacency_matrix),
create_using=nx.DiGraph)
pos = nx.spring_layout(G)
nx.draw(G, pos, with_labels=True)
```

**OBSERVATION 1**



**THOUGHT 2:** From the graph, I saw there is a path between node 0 and node 4.