Molmo and PixMo: Open Weights and Open Data for State-of-the-Art Multimodal Models

Matt Deitke[*†ψ] Christopher Clark[*†] Sangho Lee[†] Rohun Tripathi[†] Yue Yang[†]
Jae Sung Park[ψ] Mohammadreza Salehi[ψ] Niklas Muennighoff[†] Kyle Lo[†] Luca Soldaini[†]
Jiasen Lu[†] Taira Anderson[†] Erin Bransom[†] Kiana Ehsani[†] Huong Ngo[†]
YenSung Chen[†] Ajay Patel[†] Mark Yatskar[†] Chris Callison-Burch[†] Andrew Head[†]
Rose Hendrix[†] Favyen Bastani[†] Eli VanderBilt[†] Nathan Lambert[†] Yvonne Chou[†]
Arnavi Chheda[†] Jenna Sparks[†] Sam Skjonsberg[†] Michael Schmitz[†] Aaron Sarnat[†]
Byron Bischoff[†] Pete Walsh[†] Chris Newell[†] Piper Wolters[†] Tanmay Gupta[†] Kuo-Hao Zeng[†]
Jon Borchardt[†] Dirk Groeneveld[†] Jen Dumas[†] Crystal Nam[†] Sophie Lebrecht[†]
Caitlin Wittlif[†] Carissa Schoenick[†] Oscar Michel[†] Ranjay Krishna[†ψ] Luca Weihs[†]
Noah A. Smith[†ψ] Hannaneh Hajishirzi[†ψ] Ross Girshick[†ψ] Ali Farhadi[†ψ] Aniruddha Kembhavi[†ψ]

[†]Allen Institute for AI [ψ]University of Washington

Abstract

Today's most advanced multimodal models remain proprietary. The strongest open-weight models rely heavily on synthetic data from proprietary VLMs to achieve good performance, effectively distilling these closed models into open ones. As a result, the community is still missing foundational knowledge about how to build performant VLMs from scratch. We present Molmo, a new family of VLMs that are state-of-the-art in their class of openness. Our key innovation is a novel, highly detailed image caption dataset collected entirely from human annotators using speech-based descriptions. To enable a wide array of user interactions, we also introduce a diverse dataset mixture for fine-tuning that includes in-the-wild Q&A and innovative 2D pointing data. The success of our approach relies on careful choices for the model architecture details, a well-tuned training pipeline, and, most critically, the quality of our newly collected datasets, all of which will be released. The best-in-class 72B model within the Molmo family not only outperforms others in the class of open weight and data models but also compares favorably against proprietary systems like GPT-4o, Claude 3.5, and Gemini 1.5 on both academic benchmarks and human evaluation.

We will be releasing all of our model weights, captioning and fine-tuning data, and source code in the near future. Select model weights, inference code, and demo are available at https://molmo.allenai.org.

[*]Equal contribution

1. Introduction

Extensions to large language models (LLMs) that process images in addition to text have resulted in impressive multimodal capabilities, such as generating comprehensive image descriptions and accurately answering complex visual questions. The most performant of these vision-language models (VLMs), however, remain proprietary with neither model weights, data, nor code being publicly released.

With the goal of fostering scientific exploration, numerous research efforts have attempted to reproduce similar capabilities in *open* models. Early works, exemplified by LLaVA [15], produced fully open weights and training data but now lag significantly behind the state-of-the-art. More recent, stronger open-weight models have trended towards less open data: the training data may either be proprietary (*e.g.*, [5]) or, in cases where it is released, there is a heavy reliance on *synthetic* data generated by proprietary systems, *e.g.*, models are trained on datasets like ShareGPT4V [7] which uses GPT-4V [25] to generate a large set of detailed image captions. The resulting VLMs, therefore, are effectively *distillations* of proprietary VLMs, and the scientific community is still missing foundational knowledge about how to build performant VLMs from scratch.

In this work, we present the **Molmo** (**M**ultimodal **O**pen **L**anguage **Mo**del) family of state-of-the-art open VLMs with released model weights *and* released vision-language training data without any reliance on synthetic data from other VLMs, including proprietary ones. This result is achieved with a simple training pipeline in which we con-

nect an independently pre-trained, off-the-shelf vision encoder and language model and jointly train the resulting VLM to generate captions from a newly collected dataset of detailed, high-quality, dense image descriptions. After joint training, we follow standard practice and use supervised fine-tuning to produce an instruction following model. Unlike other contemporary open VLMs, we avoid multiple pre-training stages that involve freezing various parts of the model and rely on large-scale weakly paired image-text data, often three orders of magnitude larger than our high-quality data (*e.g.*, [4, 5]). The success of our approach relies on careful choices for the model architecture details, a well-tuned training pipeline, and most critically, the quality of our new datasets, collectively named **PixMo** (**Pix**els for **Mo**lmo), all of which will be released.

In practice, it is challenging to collect dense captioning datasets from human annotators. If asked to write an image description, the result often only mentions a few salient visual elements [8]. If a minimum word count is enforced, annotators will either take too long to type, making collection uneconomical, or copy-and-paste responses from proprietary VLMs, circumventing our goal of avoiding distillation. As a result, the open research community has struggled to create such datasets without relying on synthetic data from proprietary VLMs. Our key innovation is a simple but effective data collection strategy that avoids these problems: we ask annotators to describe images in *speech* for 60 to 90 seconds rather than asking them to write descriptions. We prompt the annotators to describe everything they see in great detail, including descriptions of spatial positioning and relationships. Empirically, we found that with this modality switching "trick" annotators provide far more detailed descriptions in less time, and for each description, we collect an audio receipt (*i.e.*, the annotator's recording) proving that a VLM was not used.

After training our models to generate dense captions we fine-tune them on a broad range of use cases with supervised training data. This data mixture consists of standard academic datasets as well as several newly collected datasets, including a highly diverse set of questions capturing what users in the wild might ask a model, document-focused question and answer data, analog clock reading data, and a unique new data source that grounds language in images with 2D points. This novel pointing data enables our models to answer some questions more naturally by pointing to the pixels that support the answer, improves counting accuracy (the model counts by pointing), and we believe it will open up an important future direction in which VLMs enable agents (*e.g.*, robots, web agents) to *act* by pointing in their environments, *e.g.*, to a navigation waypoint, to an object to pick up, or to a user interface button to press.

We evaluate the Molmo family of models on 11 academic benchmarks and with a human evaluation that allows us to rank models by user preference. Our most efficient model, MolmoE-1B, based on the OLMoE-1B-7B [24] mixture-of-experts LLM, nearly matches the performance of GPT-4V on both academic benchmarks and user preference. Molmo-7B-O and Molmo-7B-D, based on OLMo-7B [10] and Qwen2 7B [33], respectively, perform comfortably between GPT-4V and GPT-4o on both academic benchmarks and user preference. Our best-in-class Molmo-72B model, based on Qwen2 72B, achieves the highest academic benchmark score and ranks second by human preference, just behind GPT-4o. Our best model outperforms many state-of-the-art proprietary systems, including Gemini 1.5 Pro and Flash, and Claude 3.5 Sonnet.

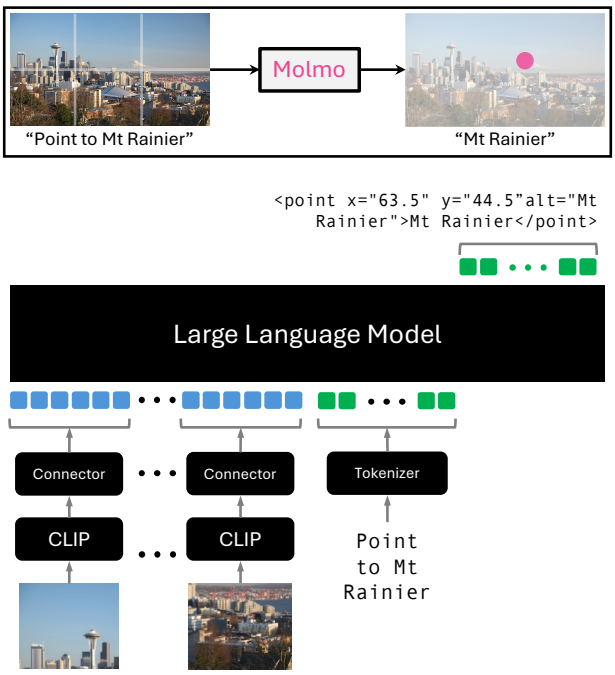

Figure 1. The **Molmo** architecture follows the simple and standard design of combining a language model with a vision encoder. Its strong performance is the result of a well-tuned training pipeline and our new **PixMo** data.

2. Architecture

Our model architecture (Figure 1) follows the simple and standard design of combining a language model with a vision encoder (*e.g.*, [15]). It consists of four components: (1) a pre-processor that converts the input image into a set of multiscale, multi-crop images, (2) a ViT image encoder that independently maps each of these images into a set of vision tokens, (3) a connector that projects the vision tokens to the language model's input dimension with an MLP and then pools the vision tokens to reduce their count, and (4) a decoder-only Transformer LLM [26, 30].

From this template, we construct a family of models that is parameterized by the choice of vision encoder and LLM.

Given these choices, the subsequent training data and recipe are the same for all models (aside from optimizer learning rates). For the vision encoder, all of our released models use OpenAI's ViT-L/14 336px CLIP model [27], which provides consistently good results (while this model uses closed data, it can be reproduced from scratch as shown by MetaCLIP [32]; we use the model from OpenAI because it was trained for higher resolution images). For the LLM, we offer a variety of choices at different scales and degrees of openness: fully open OLMo-7B-1024 (a pre-released October, 2024 backbone, which will be released at a later date), fully open OLMoE-1B-7B (our most efficient model), open-weight Qwen2 7B, and open-weight Qwen2 72B (our best-performing model).

3. Data and Training

Starting from an independently pre-trained vision encoder and LLM, our training processing is simple and consists of only two stages: (1) multimodal pre-training for caption generation using **PixMo-Cap**, our newly collected caption data and (2) supervised fine-tuning using a mixture of academic datasets and our newly collected supervised **PixMo-⋆** family of datasets. All model parameters are updated in both stages. We do not use RLHF.

Stage 1: Caption generation. In this stage, we join the vision encoder and LLM with our randomly initialized connector and train all model parameters on the task of caption generation. We collected the **PixMo-Cap** training data for this stage as follows.

We started by sourcing web images according to a diverse set of ∼70 high-level topics (*e.g.*, street signs, memes, food, drawings, websites, blurry photos, *etc.*), and for each image we asked three annotators to describe the image in detail by speaking for at least 60 seconds (in later stages of collection we increased this to 90 seconds and used a single annotator per image; we found this was more efficient without a loss in quality). The annotators were prompted with a list of simple questions to answer in their descriptions:

- *What is the image at first glance?*
- *What are the objects and their counts?*
- *What does the text say?*
- *What are the positions of the objects?*
- *What subtle details are noticeable?*
- *What is in the background?*
- *What is the style and color?*

The annotators' audio was then transcribed using an off-the-shelf speech-to-text system, and then the transcribed text was processed using a language-only LLM to improve the text quality (*e.g.*, removing spoken artifacts, normalizing style). We also created a fourth image description by asking the language-only LLM to summarize the three original transcripts into a single description.

Our training process uses all four of these image LLM-processed transcripts, when available, as a form of naturalistic data augmentation. In total, we trained on 712k distinct images with ∼1.3M captions (including the augmentation).

Stage 2: Supervised fine-tuning. After training for captioning, we fine-tune all model parameters on a mixture of supervised training data. This mixture includes common academic datasets and several new **PixMo** datasets, described next.

- **PixMo-AskModelAnything**: We collected this data with the goal of enabling the model to answer a diverse set of questions covering what users might ask it when deployed in the wild. To create image-question-answer triplets, we had annotators work with a language-only LLM. First, an annotator would select an image from a large pool and then write a question about it. We used our stage 1 model to generate a dense caption for the image and passed that caption, OCR output for the image (from a non-VLM, off-the-shelf OCR model), and the question to a language-only LLM. The LLM provided an answer (emphasizing again that it had no access to the image), which the annotator could either accept or reject. If rejected, they would describe what was wrong with the answer and ask the LLM to fix it. The annotator iterated this process until the answer was acceptable. For some of the data, we asked annotators to ask questions following a specific prompt, including unusual requests such as asking for the answer to be written upside down (which is possible with Unicode). This dataset has 162k question-answer pairs and 73k images.
- **PixMo-Points**: We collected pointing data that achieves three goals: (1) enables the model to point to anything described by text, (2) enables the model to count by pointing, and (3) enables the model to use pointing as a natural form of visual explanation when answering questions. To collect data for the first two goals, we asked human annotators to point at something in an image, write a description of it, and then point to every instance of it in the image (making the pointing exhuastive). We also collected "not present" data so models can learn to respond appropriately when asked about something *not* in the image. This data also naturally allows us to train the model to answer counting questions with points acting as a form of chain-of-thought. We collected 2.3M question-point pairs from 428k images. To enable points as a form of explanation, we followed the PixMo-AskModelAnything pipeline but augmented it so that the annotator could pass the LLM a list of text-annotated points. The LLM was then prompted to use these points, if appropriate, to support its answer. We collected 79k question-answer pairs from 29k images.
- **PixMo-CapQA**: We generated an additional 214k question-answer pairs from 165k images by prompting

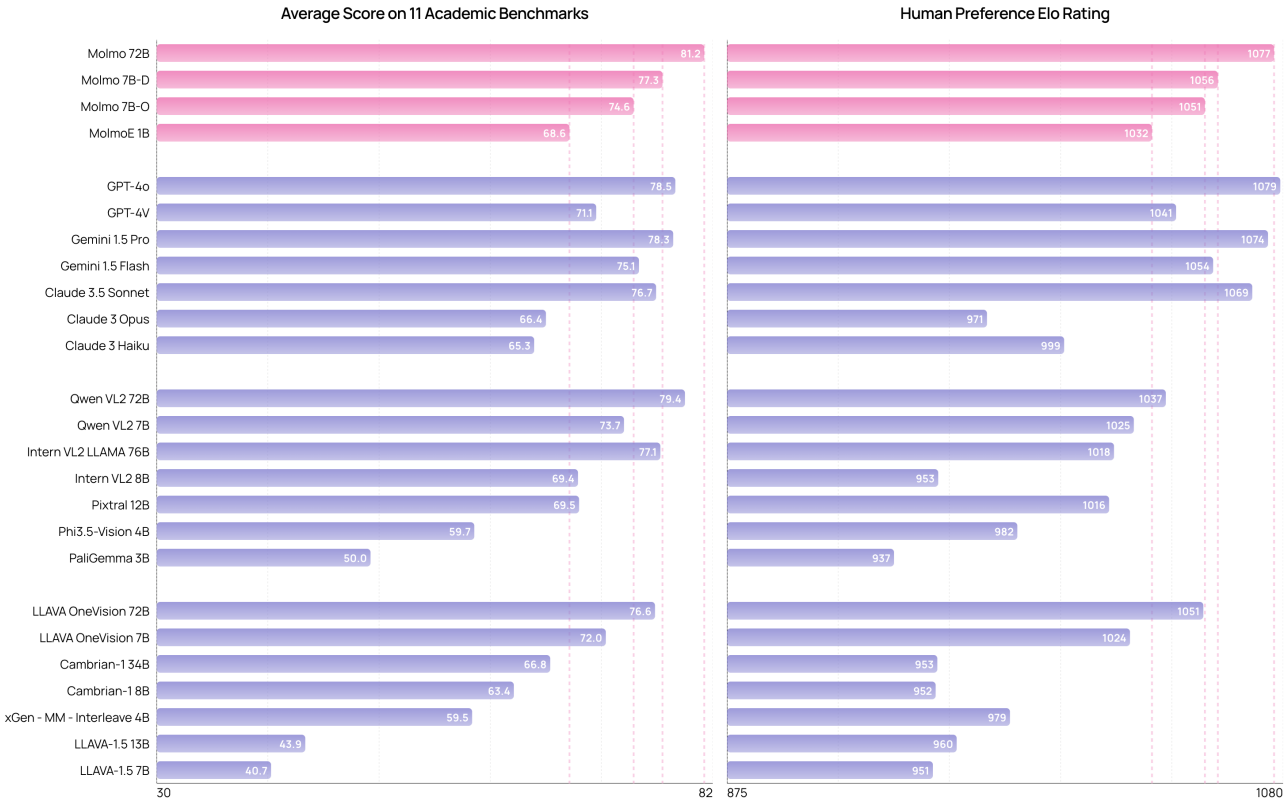

Figure 2. (Left) Average scores on the 11 academic benchmarks. See Table 1 for per-benchmark results. (Right) Elo ratings from our human preference evaluation.

a language-only LLM to ask and answer questions given only the ground-truth caption for an image. To increase diversity, we created a list of high-level topics and styles and asked the model to use them.

- **PixMo-Docs**: We prompted an LLM to generate code for 255k text and figure-heavy images, including charts, documents, tables, and diagrams. We then prompted the LLM to generate 2.3M question-answer pairs based on privileged access to the code (the images were not used).
- **PixMo-Clocks**: We constructed a new dataset of synthetic analog clocks with questions and answers about the time. The images were rendered from ~50 different watches and a diverse set of ~160k realistic watch face styles featuring randomly chosen times. We collected 826k examples.
- **Academic datasets**: VQA v2 train (COCO 2014 subset) [9], TextVQA train [29], OK-VQA train [19], ChartQA train (human and augmented examples balanced equally) [20], DocVQA train [21], InfographicVQA train [22], AI2D train (transparent and opaque label boxes) [13], A-OKVQA train [28], AndroidControl train [14], ScienceQA train [16], TabMWP train [17], ST-VQA train [6], TallyQA train [3], DVQA train [11], FigureQA train [12], and PlotQA train [23].

4. Evaluation

Vision-language model evaluation is evolving rapidly, with new academic benchmarks constantly appearing. These benchmarks work well for evaluating specific skills, but doing well on them often requires answering questions in a benchmark-specific style. These answers are often short and do not work well in a conversational setting. As a result, academic benchmarks provide only a partial picture of how a model performs. To complement these benchmarks, we perform a human evaluation that allows us to rank models according to user preference.

For academic benchmarking, we attempted to collect results for all models on a set of 11 commonly used academic benchmarks.[1] We prioritized using numbers published by the authors themselves when they were available, but many were missing. When results were not available, we attempted to find the best previously reported values from other technical reports or from public leaderboards, such as the OpenVLM Leaderboard. Finally, if a value was still

[1] AI2D test, ChartQA test, VQA v2 test, DocVQA test, InfographicVQA test, TextVQA val, RealWorldQA [2], MMMU val [34], MathVista testmini [18], CountBenchQA [5], Flickr Count (we collected this new dataset that is significantly harder than CountBenchQA).

		VLM		LLM Backbone		Vision Encoder	
Category	Model	Open Weights	Open Data + Code	Open Weights	Open Data + Code	Open Weights	Open Data + Code
Molmo	Molmo-72B	Open	Open	Open	Closed	Open	Closed
	Molmo-7B-D	Open	Open	Open	Closed	Open	Closed
	Molmo-7B-O	Open	Open	Open	Open	Open	Closed
	MolmoE-1B	Open	Open	Open	Open	Open	Closed
API Models	GPT-4o	Closed	Closed	Closed	Closed	Closed	Closed
	GPT-4V	Closed	Closed	Closed	Closed	Closed	Closed
	Gemini 1.5 Pro	Closed	Closed	Closed	Closed	Closed	Closed
	Gemini 1.5 Flash	Closed	Closed	Closed	Closed	Closed	Closed
	Claude 3.5 Sonnet	Closed	Closed	Closed	Closed	Closed	Closed
	Claude 3 Opus	Closed	Closed	Closed	Closed	Closed	Closed
	Claude 3 Haiku	Closed	Closed	Closed	Closed	Closed	Closed
Open Weights	Qwen VL2 72B	Open	Closed	Open	Closed	Open	Closed
	Qwen VL2 7B	Open	Closed	Open	Closed	Open	Closed
	Intern VL2 LLAMA 76B	Open	Closed	Open	Closed	Open	Closed
	Intern VL2 8B	Open	Closed	Open	Closed	Open	Closed
	Pixtral 12B	Open	Closed	Open	Closed	Open	Closed
	Phi3.5-Vision 4B	Open	Closed	Open	Closed	Open	Closed
	PaliGemma 3B	Open	Closed	Open	Closed	Open	Closed
Open Weights & Data	LLAVA OneVision 72B	Open	Distilled	Open	Closed	Open	Closed
	LLAVA OneVision 7B	Open	Distilled	Open	Closed	Open	Closed
	Cambrian-1 34B	Open	Distilled	Open	Closed	Open	Closed
	Cambrian-1 8B	Open	Distilled	Open	Closed	Open	Closed
	xGen - MM - Interleave 4B	Open	Distilled	Open	Closed	Open	Closed
	LLAVA-1.5 13B	Open	Open	Open	Closed	Open	Closed
	LLAVA-1.5 7B	Open	Open	Open	Closed	Open	Closed

Figure 3. **VLM Openness Comparison.** We characterize the openness of VLMs based on two attributes (open weights, open data and code) across three model components (the VLM and its two pre-trained components, the LLM backbone and the vision encoder). In addition to open vs. closed, we use the "distilled" label to indicate that the data used to train the VLM includes images and text generated by a different, proprietary VLM, meaning that the model cannot be reproduced without a dependency on the proprietary VLM.

missing, we computed it ourselves. We note that computing results is difficult in practice. For a fixed model, results on a given benchmark can vary by a large amount (*e.g.*, 10 percentage points) depending on the details of how it was evaluated. Further complicating matters, in many cases, critical evaluation details, such as what prompts were used or how the data was processed, may not be available, making it difficult to reproduce published results. These issues underscore the importance of open evaluation.

We also avoid making a strong distinction between claimed "zero-shot" performance (often reported for closed-data models) and the supervised performance of models that explicitly train on benchmark training sets. The distinction between supervised training and zero-shot transfer is fuzzy since one can curate new data sources that serve as effective proxies for any given benchmark's literal training data. When training data is not disclosed, the community has no means of evaluating zero-shot transfer claims.

For our human evaluation, we collected a diverse set of 15k image and text prompt pairs and queried a set of VLMs

Model	AI2D test	ChartQA test	VQA v2 testdev	DocVQA test	Info. VQA test	TextVQA val	RealWorldQA	MMMU val	MathVista testmini	CountBenchQA	Flickr Count	Average
API call only												
GPT-4V	89.4	78.1	77.2	87.2	75.1	78.0	61.4	63.1	58.1	69.9	45.0	71.1
GPT-4o-0513	94.2	85.7	78.7	92.8	79.2	77.4	75.4	**69.1**	63.8	87.9	59.6	78.5
Gemini 1.5 Flash	91.7	85.4	80.1	89.9	75.3	78.7	67.5	56.1	58.4	81.6	61.1	75.1
Gemini 1.5 Pro	94.4	87.2	80.2	93.1	81.0	78.7	70.4	62.2	63.9	85.8	64.3	78.3
Claude-3 Haiku	86.7	81.7	68.4	88.8	56.1	67.3	45.5	50.2	46.4	83.0	43.9	65.3
Claude-3 Opus	88.1	80.8	66.3	89.3	55.6	67.5	49.8	59.4	50.5	83.6	43.3	66.7
Claude-3.5 Sonnet	94.7	**90.8**	70.7	95.2	74.3	74.1	60.1	68.3	**67.7**	89.7	58.3	76.7
Open weights only												
PaliGemma-mix-3B	72.3	33.7	76.3	31.3	21.4	56.0	55.2	34.9	28.7	80.6	60.0	50.0
Phi3.5-Vision-4B	78.1	81.8	75.7	69.3	36.6	72.0	53.6	43.0	43.9	64.6	38.3	59.7
Qwen2-VL-7B	83.0	83.0	82.9	94.5	76.5	84.3	70.1	54.1	58.2	76.5	48.0	73.7
Qwen2-VL-72B	88.1	88.3	81.9	**96.5**	84.5	**85.5**	**77.8**	64.5	70.5	80.4	55.7	79.4
InternVL2-8B	83.8	83.3	76.7	91.6	74.8	77.4	64.2	51.2	58.3	57.8	43.9	69.4
InternVL2-LLaMa-3-76B	87.6	88.4	85.6	94.1	**82.0**	84.4	72.7	58.2	65.5	74.7	54.6	77.1
Pixtral-12B	79.0	81.8	80.2	90.7	50.8	75.7	65.4	52.5	58.0	78.8	51.7	69.5
Open weights + data († distilled)												
LLaVA-1.5-7B	55.5	17.8	78.5	28.1	25.8	58.2	54.8	35.7	25.6	40.1	27.6	40.7
LLaVA-1.5-13B	61.1	18.2	80.0	30.3	29.4	61.3	55.3	37.0	27.7	47.1	35.2	43.9
xGen-MM-interleave-4B†	74.2	60.0	81.5	61.4	31.5	71.0	61.2	41.1	40.5	81.9	50.2	59.5
Cambrian-1-8B†	73.0	73.3	81.2	77.8	41.6	71.7	64.2	42.7	49.0	76.4	46.6	63.4
Cambrian-1-34B†	79.7	75.6	83.8	75.5	46.0	76.7	67.8	49.7	53.2	75.6	50.7	66.8
LLaVA OneVision-7B†	81.4	80.0	84.0	87.5	68.8	78.3	66.3	48.8	63.2	78.8	54.4	72.0
LLaVA OneVision-72B†	85.6	83.7	85.2	91.3	74.9	80.5	71.9	56.8	67.5	84.3	60.7	76.6
The Molmo family: Open weights, Open data, Open training code, Open evaluations												
MolmoE-1B	86.4	78.0	83.9	77.7	53.9	78.8	60.4	34.9	34.0	87.2	79.6	68.6
Molmo-7B-O	90.7	80.4	85.3	90.8	70.0	80.4	67.5	39.3	44.5	89.0	83.3	74.6
Molmo-7B-D	93.2	84.1	85.6	92.2	72.6	81.7	70.7	45.3	51.6	88.5	84.8	77.3
Molmo-72B	**96.3**	87.3	**86.5**	93.5	81.9	83.1	75.2	54.1	58.6	**91.2**	**85.2**	**81.2**

Table 1. Academic benchmark results covering ten commonly used datasets plus one newly collected counting benchmark, Flickr Count, which focuses on counting in more challenging natural images than CountBenchQA. We organize models into four groups: (top) proprietary models that can only be accessed through API calls, (upper middle) models with released weights but closed data, (lower middle) models with released weights and released training data, noting that some of these distill (†) from other models by training on synthetic data generated by proprietary VLMs, and (bottom) the Molmo family of models.

for responses. We then sampled and presented the resulting image-text-response triplets for all VLM pairings to a set of ~870 human annotators who gave pairwise preference rankings. Across all pairs of models, we collected greater than 325k preference ratings (~450 matches per model pair). From these preference rankings, we calculated an Elo ranking using the Bradley-Terry model following the methodology of LMSYS Org's Chatbot Arena [1].

Broadly speaking, the academic benchmark results and human evaluation strongly agree, with the exception of Qwen2-VL [31], which performs strongly on the academic benchmarks and comparatively underperforms in the human evaluation. We highlight a few key results:

- Our most efficient model, MolmoE-1B, based on the OLMoE-1B-7B mixture-of-experts LLM, nearly matches the performance of GPT-4V on both academic benchmarks and Elo.
- Our OLMo-7B-1024 and Qwen2 7B based models perform comfortably between GPT-4V and GPT-4o on both academic benchmarks and the Elo ranking.
- Our best-in-class Qwen2 72B based model achieves the highest academic benchmark score and ranks second in Elo, just behind GPT-4o.
- Our best model outperforms many state-of-the-art proprietary systems, including Gemini 1.5 Pro and Flash and Claude 3.5 Sonnet.
- To highlight Molmo's potential for *action* we tested Molmo-72B on AndroidControl [14] where it achieved 88.7% low-level accuracy and 69.0% high-level accuracy, comparing well to the results of 83.2% and 70.8% reported in [14].

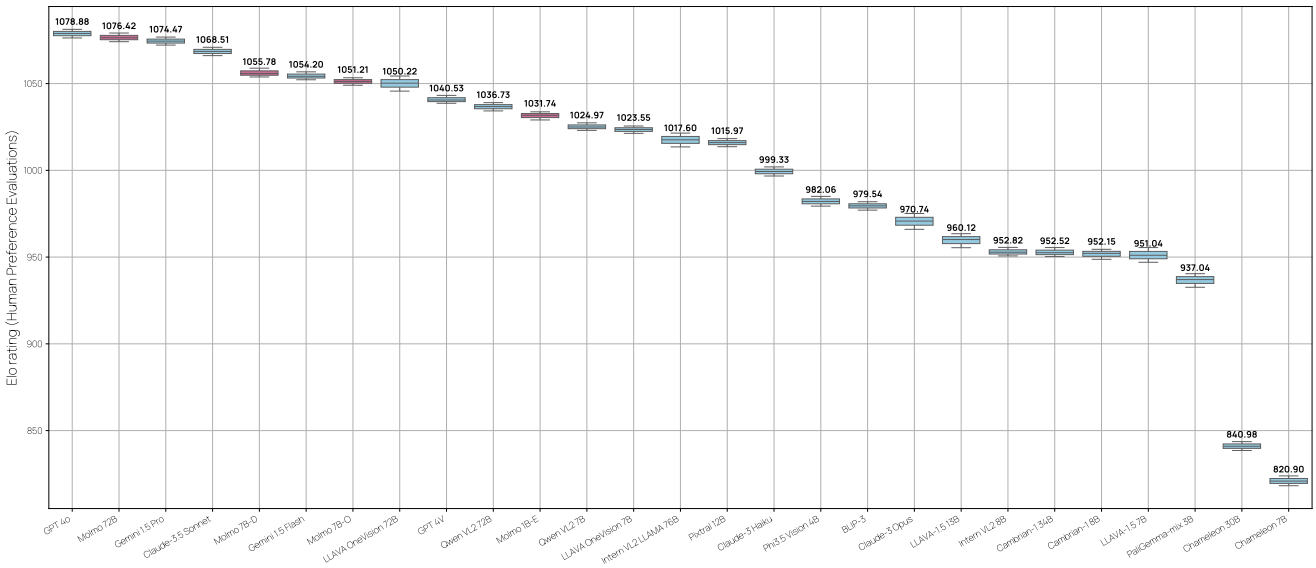

Figure 4. Our Elo human preference evaluations used 15k image and text prompt pairs. We queried each VLM for responses, and presented the resulting image-text-response triplets for all VLM pairings to a set of ~870 human annotators who gave pairwise preference rankings, for a total of 325k pairwise comparisons across 27 models, making it the biggest human preference evaluation for multimodal models to date. As a reference, our ELO rankings are based on 3× more votes than Chatbot Arena (LMSYS) for vision models.

5. Release Plan

Our first release on September 25, 2024 includes a demo, inference code, and the following model weights:

- MolmoE-1B using the fully open OLMoE-1B-7B mixture-of-experts LLM
- Molmo-7B-O using the fully open OLMo-7B-1024 LLM (an October 2024 pre-release, to be public later)
- Molmo-7B-D, our demo model, using the open-weight Qwen2 7B LLM
- Molmo-72B, our best performing model, using the open-weight Qwen2 72B LLM

Building upon this work, soon we'll be releasing:

- A more detailed version of this technical report
- All PixMo datasets
- Updated model weights
- Training and evaluation code

References

[1] Chatbot arena: New models and Elo system update. https://lmsys.org/blog/2023-12-07-leaderboard/. Accessed: 2024-09-24. 6

[2] RealWorldQA. https://huggingface.co/datasets/xai-org/RealworldQA. Accessed: 2024-09-24. 4

[3] Manoj Acharya, Kushal Kafle, and Christopher Kanan. TallyQA: Answering complex counting questions. In *AAAI*, 2019. 4

[4] Meta AI. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 2

[5] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, Julian Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bošnjak, Xi Chen, Matthias Minderer, Paul Voigtlaender, Ioana Bica, Ivana Balazevic, Joan Puigcerver, Pinelopi Papalampidi, Olivier Henaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiaohua Zhai. PaliGemma: A versatile 3B VLM for transfer. *arXiv preprint arXiv:2407.07726*, 2024. 1, 2, 4

[6] Ali Furkan Biten, Ruben Tito, Andres Mafla, Lluis Gomez, Marçal Rusinol, Ernest Valveny, CV Jawahar, and Dimosthenis Karatzas. Scene text visual question answering. In *ICCV*, 2019. 4

[7] Lin Chen, Jisong Li, Xiaoyi Dong, Pan Zhang, Conghui He, Jiaqi Wang, Feng Zhao, and Dahua Lin. ShareGPT4V: Improving large multi-modal models with better captions. *arXiv preprint arXiv:2311.12793*, 2023. 1

[8] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015. 2

[9] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *CVPR*, 2017. 4

[10] Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, A. Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Daniel Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hanna Hajishirzi. OLMo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*, 2024. 2

[11] Kushal Kafle, Brian Price, Scott Cohen, and Christopher Kanan. DVQA: Understanding data visualizations via question answering. In *CVPR*, 2018. 4

[12] Samira Ebrahimi Kahou, Vincent Michalski, Adam Atkinson, Ákos Kádár, Adam Trischler, and Yoshua Bengio. FigureQA: An annotated figure dataset for visual reasoning. *arXiv preprint arXiv:1710.07300*, 2017. 4

[13] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *ECCV*, 2016. 4

[14] Wei Li, William Bishop, Alice Li, Chris Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the effects of data scale on computer control agents. *arXiv preprint arXiv:2406.03679*, 2024. 4, 6

[15] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 1, 2

[16] Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *NeurIPS*, 2022. 4

[17] Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. In *ICLR*, 2023. 4

[18] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *ICLR*, 2024. 4

[19] Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *CVPR*, 2019. 4

[20] Ahmed Masry, Do Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *ACL*, 2022. 4

[21] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. DocVQA: A dataset for VQA on document images. In *WACV*, 2021. 4

[22] Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. InfographicVQA. In *WACV*, 2022. 4

[23] Nitesh Methani, Pritha Ganguly, Mitesh M Khapra, and Pratyush Kumar. PlotQA: Reasoning over scientific plots. In *WACV*, 2020. 4

[24] Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, Yuling Gu, Shane Arora, Akshita Bhagia, Dustin Schwenk, David Wadden, Alexander Wettig, Binyuan Hui, Tim Dettmers, Douwe Kiela, Ali Farhadi, Noah A. Smith, Pang Wei Koh, Amanpreet Singh, and Hannaneh Hajishirzi. OLMoE: Open mixture-of-experts language models. *arXiv preprint arXiv:2409.02060*, 2024. 2

[25] OpenAI. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1

[26] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. *OpenAI Blog*, 2018. 2

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 3

[28] Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. A-OKVQA: A benchmark for visual question answering using world knowledge. In *ECCV*, 2022. 4

[29] Amanpreet Singh, Vivek Natarjan, Meet Shah, Yu Jiang, Xinlei Chen, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *CVPR*, 2019. 4

[30] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2

[31] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 6

[32] Hu Xu, Saining Xie, Xiaoqing Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying CLIP data. In *ICML*. 3

[33] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024. 2

[34] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *CVPR*, 2024. 4