Evaluating 'Graphical Perception' with Multimodal LLMs

Rami Huu Nguyen*
University of Massachusetts Boston

Kenichi Maeda†
University of Massachusetts Boston

Mahsa Geshvadi‡
University of Massachusetts Boston

Daniel Haehn§
University of Massachusetts Boston

1 Initial Experiment

Before we started our official experiments, our study investigated how MLLMs pretrained models perform for three types of images in white and black backgrounds. We used one of the random tasks, which is angle, to answer this question. To make everything consistent for our experiments, we first generated 55 images with varying angles and directions. We then added black and white backgrounds for three types of images, including aliased, anti-aliased, and vectorized. Thus, in the end, we compared six different types of images and computed the average MLAE to compare their performances.

Figure 1 recorded that the aliased image with a black background achieved the lowest total average of MLAE among all models at **15.82**, followed by the aliased image with white background at **15.83**, the anti-aliased image with a black background at **15.77**, the anti-aliased image with white background at **16.45**, the vectorized image with a black background at **16.21**, and the vectorized image with white background at **16.20**. Therefore, we used the aliased image with a black background for our study.

To answer why our MLLMs have a wider error bar, as discussed in our main paper, we sum our ground truth and prediction responses into a list of unique values. This grouping aims to explore outliers for both fine-tuned and pretrained models. From this outlier analysis, our study might explore the reason for a more expansive error bar.

1.1 Experiment 1

Figure 2 presents significant outliers for pre-trained and fine-tuned models in all tasks. Some key examples of outliers are:

- **In angle tasks**, our answer ranges are expected between 0 and 90, but ***fine-tuned models*** produce some responses with values over 100, and there is one extreme outlier at a value of 500.
- **In length tasks,** the range of our answer is forecasted to lie between 0 and 100. Still, **fine-tuned models** return a particular value above the range, with one outlier hitting 175.
- **In area tasks**, the expected answer falls between 0 and 5026.5, but ***all pretrained models*** generated various responses exceeding the range, including one extreme case at over 60,000.
- In **volume tasks**, although the anticipated range for our answers is between 0 and 800, some responses surpass this range, with one outlier hitting 30,000,000.

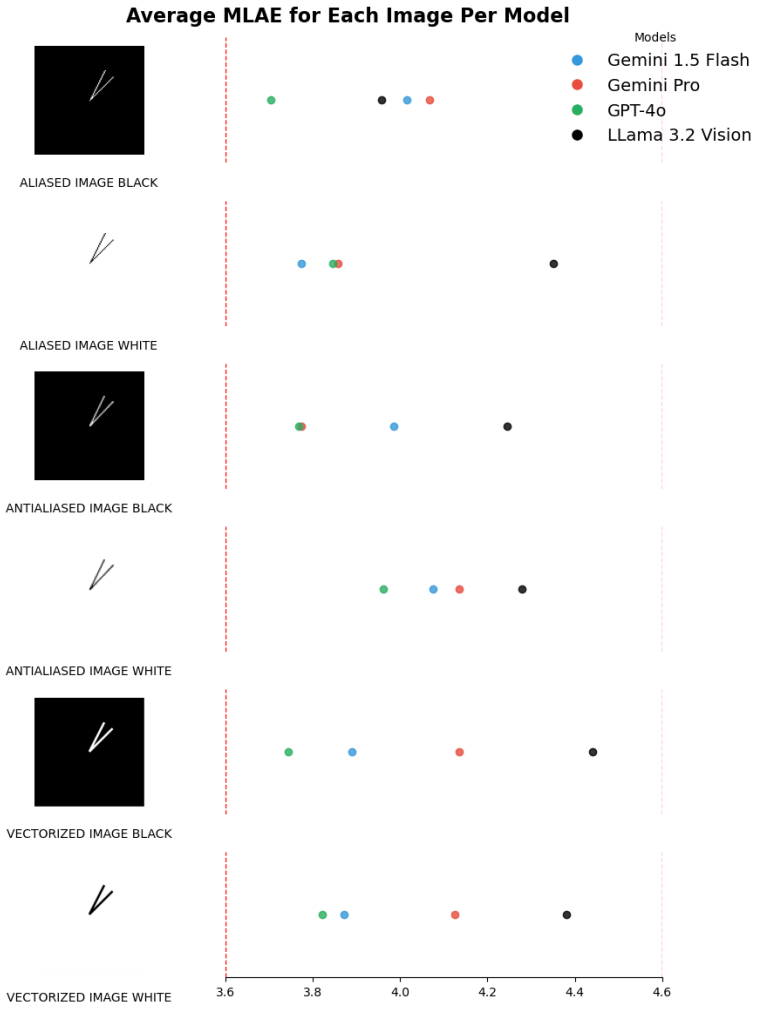

Figure 1: Comparison of Average MLAE across models for different image types.

*e-mail: rami@mpsych.org
†e-mail: kenichi.maeda001@umb.edu
‡e-mail: mahsa.geshvadi001@umb.edu
§e-mail: daniel.haehn@umb.edu

These outliers might explain the wide range of error bars for all models. We prompted our MLLMs to provide a specific answer range, but they still produced several answers that were outside the range. These outliers suggest that zero-shot prompting does not work consistently across the nine tasks for all pretrained models except for curvature tasks.

1.2 Experiment 2

Figure 3 highlights all unique values ranging from 0 to 83; in this range, we expected our answers to be between 0 and 1. Looking at all pretrained models, they have a few responses for two tasks starting from 1 to 40 and one extreme outlier at 83.

Overall, fine-tuned and pretrained models exhibit a similar distribution for unique counts below 250. However, there are several notable spikes in unique counts for pretrained models, for instance, for some responses 0.2, 0.4, 0.6 for **pie charts and bar charts**. Also, fine-tuned models prefer specific values such as 0.78 and 0.81, with unique counts of 268 and 371, respectively. These uneven response spikes and outliers might cause a wider spread of error bars.

1.3 Experiment 3

We expected our models to provide answers ranging from 0 to 1; nonetheless, our pretrained models displayed a wider range of unique values, and some of the responses were over 1, defined as outliers. On the other hand, our fine-tuned models' answers are within the range.

Moreover, we found a starting point of 200 counts for unique responses across all models, indicating all models produced specific values more frequently than usual. Key examples include:

- Fine-tuned models: 235 responses at 0.39 and 520 responses at 0.32.
- Pretrained models: 280 responses at 0.80 and 387 responses at 0.75.

This suggests both fine-tuned and pretrained models prefer specific outputs often and potentially the cause of the wider MLAE error. Figure 4 showcases our fine-tuned models (purple), showing a wider error distribution in MLAE values across **all task types** compared to pretrained models.

1.4 Experiment 4

Figure 5 emphasizes the distribution of each unique value for **framed and unframed tasks** across pretrained and fine-tuned models. For pretrained models, we expected our models to answer between 49 and 60; however, they have various responses out of the range below 49 and after 60. Also, some unique answers have over 350 counts between 49 and 60. Those out-of-range predictions introduce variability and likely contribute to a wider MLAE error bar for both tasks.

Although fine-tuned models provide answers within our defined range of 0 to 60, in the middle plot of Figure 5, our fine-tuned models prefer to respond at various values, at 51, 56, 57, and 58, more often than the other values. These images linked to each value (51, 56, 57, 58) might look very similar, and thus, our fine-tuned models potentially have the challenge of regressing value from those images. Since the fine-tuned models potentially struggle to distinguish slight differences in the images, our fine-tuned models might stick to those values to guess their values.

1.5 Experiment 5

Our answer ranges for each task lie between 0 and 10; nonetheless, some outliers from pretrained models were detected in the value range of 11 to 7000, especially for the **1000-dots** tasks, and a few responses for the **10-dots** tasks, causing larger error bars. Moreover, for **100-dot** tasks, our pretrained models prefer responses at 7, and there are over 1000 unique counts for this answer.

Looking more closely at fine-tuned models, the distribution is generally uneven for all tasks, even though they have a similar count of each unique prediction value. Sometimes, predictions are either higher or lower than the ground truth, which leads to significant errors. Figure 6 implies that pre-trained and fine-tuned models might not fully generalize well in these tasks involving more complex patterns.

2 Data Preparation and Experimental Setup

2.1 Data Generation

Dataset Size Per Experiment:

- **Training:** 5,000 images
- **Validation:** 1,000 images
- **Testing:** 55 images per task

Data Generation Process:

- Following Haehn et al., stimuli were randomly generated and added until reaching the target dataset size.
- Each dataset had unique random parameters, ensuring no data leakage between training, validation, and test sets.

Testing Procedure:

- Each model was tested **three times per task** using the same **55 test images** to ensure fairness.
- This setup resulted in **825 total rows** in the dataset (**55 images × 5 models × 3 repetitions per model**).

Noise Injection & Zero-Shot Prompting:

- **5% noise** was added to images to challenge model robustness.
- **Zero-shot prompting techniques** were used to evaluate model performance across all experiments.

2.2 Handling invalid responses and dataset balancing

Initially, our study collected 825 responses per task across all five experiments, with each experiment has three runs. However, invalid and missing responses were found, primarily for pretrained models from E1 to E5 in each experiment. To ensure fair comparisons between models, we excluded these invalid responses. However, since each model produced a different number of invalid responses, this resulted in an imbalance in the valid response count per model. As these invalid responses vary in each run, we balanced the dataset by randomly choosing the global minimum number of valid responses.

Examples of Invalid Responses by Experiment:

- **E1:**
  - *I can't determine the direction based on this image.*
  - *I'm unable to visually determine the curvature of a Bezier curve from an image.*

  - *I'm unable to provide a curvature estimate from the image.*
  - *I'm unable to determine the curvature from the image provided.*

- **E2:**
  - *To answer this question accurately, I need to see the pie chart.*
  - *I'm unable to view the pie chart, but if you provide a description of the segments' sizes or their relative proportions, I can help guide you through the calculation.*
  - *I'm sorry, I can't analyze or identify specific parts of an image including pie chart segments. Please provide more details or specify the values in the list for further assistance.*
  - *I apologize, but I can't determine the segments of the pie chart based on the image provided.*

- **E3:**
  - *I can't view the image. Please provide a description or another question.*
  - *I'm unable to view the image clearly. Could you provide a clearer version or describe the chart?*
  - *I can't analyze the image directly. Could you please provide more details or describe the chart?*
  - *I can't view or interpret details in the image, including marked features.*

- **E4:**
  - *I'm unable to view the image provided. Could you please describe it or try another format?*
  - *I'm sorry. I can't help with that.*
  - *The average length of the two bars is 49 pixels.*
  - *The length of the right bar is approximately 60 pixels.*

- **E5:**
  - *"No response."*

We include an example of E1 below to demonstrate our approach:

Example (E1):

Each task originally had 825 responses, but valid response counts varied between 812 and 825 for all runs. To address this, we randomly removed valid responses to match the global minimum number of valid responses across models per run, ensuring dataset balance for an unbiased evaluation. For instance, in our E1, we have 9 tasks, and each originally has 825 responses. Nonetheless, the number of valid responses per task varies between 812 and 825 due to invalid responses each run. Since these invalid responses vary in each run, we balanced the dataset by randomly choosing the global minimum number of valid responses (e.g. 812).

2.3 Unique labels

Each experiment involves a dataset comprising a combination of unique labels across its subsets.

E1:

- Training datasets: 240 unique labels.
- Validation datasets: 220 unique labels.
- Test datasets: 220 unique labels.

E2:

- Training datasets: 7780 unique labels.
- Validation datasets: 1880 unique labels.
- Test datasets: 970 unique labels.

E3:

- Training/validation/test datasets contain 38 unique labels.

E4:

- Training, validation, and test datasets: 132 unique labels each.

E5:

- Dataset comprises a combination of unique labels across subsets.
- Training, validation, and test datasets: 10 unique labels each.

3 Fine-tuning Details

For fine-tuned MLLMs, we used Llama 3.2 Vision, with 6.0 billion parameters, of which 94.4 million were trainable. We also produced 15 fine-tuned MLLMs (each experiment has three fine-tuned MLLMs) for this study. Our fine-tuned MLLMs using a learning rate of 0.0001, a weight decay of 0.01, a batch size of 2, and 5 epochs.

Additionally, we utilized Parameter-Efficient Fine-Tuning (PEFT) and configured LoRA (Low-Rank Adaptation) with the following settings: a LoRA alpha of 256, a dropout rate of 0.1, and a rank of 128. The bias was set to "none," targeting the "q proj" and "v proj" modules, and the task type was specified as feature extraction. We also used 4-bit precision, NormalFloat(nf4), and bfloat16 to save memory and achieve greater accuracy.

4 Zero-shot prompts

4.1 Experiment 1

Our zero-shot prompts are for our elementary perceptual experiment:

- **Position Common Scale**: Estimate the block's vertical position (range: 0-60, top to bottom). Number only. No explanation.
- **Position Non-Aligned Scale**: Estimate the block's vertical position (range: 0-60, top to bottom). Number only. No explanation.
- **Length**: Estimate the line length from top to bottom (range: 0-100). Number only. No explanation.

- **Direction**: Estimate the line's direction (range: 0-359 degrees). Number only. No explanation.
- **Angle**: Estimate the angle (range: 0-90 degrees). Number only. No explanation.
- **Area**: Estimate the area of a circle, ensuring your answer falls within the range of 3.14 to 5026.55 square units. Assume the circle fits within a 100x100 pixel image. Provide only the numeric value, no explanation.
- **Volume**: Estimate the volume of a cube, with your answer restricted to the range of 1 to 8000 cubic units. Assume the cube fits within a 100x100 pixel image. Provide only the numeric value, no explanation.
- **Curvature**: Estimate the line curvature (range: 0.000 to 0.088) of a Bezier curve constrained within a 100x100 pixel space. Provide only the numeric curvature value (up to 3 decimal places), no explanation.
- **Shading**: Estimate shading density (range: 0-100). Number only. No explanation.

4.2 Experiment 2

Our zero-shot prompts for the position-angle experiment are:

Both bar and pie chart have separated prompt as mentioned in this material; nevertheless, they mostly have similar zero-shot prompting instructions:

The pie or bar chart you are looking at is created as follows:

- First, create a list of five values, each between 3 and 39, and all values add up to 100.
- Next, divide each value in the list by the largest value so that the largest value becomes 1.0.
- Now, look at the pie chart again.
- Identify the largest segment, which is marked with a dot.
- Estimate the ratio of the other four values to maximum.
- Format your answer as [1.0, x.x, x.x, x.x, x.x].

The difference between the two task prompts is how all models estimate the ratio of the other four values to maximum:

- **Pie chart:** Go counterclockwise around the pie starting from the largest segment, estimating the ratio of the other four values to the maximum.
- **Bar chart:** Move left to right along the bar chart starting from the largest bar, estimating the ratio of the other four values to the maximum.

4.3 Experiment 3

Our zero-shot prompts for all five tasks for position-length experiment are as follows:

Type 1, Type 2, and Type 3 have a similar prompt structure, with the only differences being the chart types: grouped, divided, and mixed bar charts. The prompt is:

- *In the grouped/divided/mixed bar chart, compare the heights of the two marked bars.*
- *Estimate the ratio of the height of the shorter marked bar to the height of the taller marked bar.*
- *Use a scale from 0 to 1, where 1 indicates that both marked bars are of equal height. No explanation.*

Type 4 and Type 5 also have a similar prompt structure, with the distinction being between divided stacked bars and the left bar of the mixed divided stacked bar chart. The prompt is:

- *In the divided stacked bars or the left bar of the mixed divided stacked bar chart, compare the lengths of the two marked segments in the left and right bars.*
- *Estimate the ratio of the shorter marked segment's length to the length of the taller marked segment.*
- *Use a scale from 0 to 1, where 1 indicates equal length. No explanation.*

4.4 Experiment 4

Our zero-shot prompting for our bars and rectangles experiment:

- *Estimate the lengths of the two **framed and without framed bars**. Both lengths should fall between 49 and 60 pixels. No explanation. Format of the answer [xx, xx].*

4.5 Experiment 5

Our research also includes zero-shot prompts for our point cloud experiment. Our prompts are:

- **Task 10**: Please estimate how many dots were added to the initial 10 dots. The answer must be within the range of 1 to 10. Number only. No explanation.
- **Task 100**: Please estimate how many dots were added to the initial 100 dots. The answer must be within the range of 1 to 10. Number only. No explanation.
- **Task 1000:** Please estimate how many dots were added to the initial 1000 dots. The answer must be within the range of 1 to 10. Number only. No explanation.

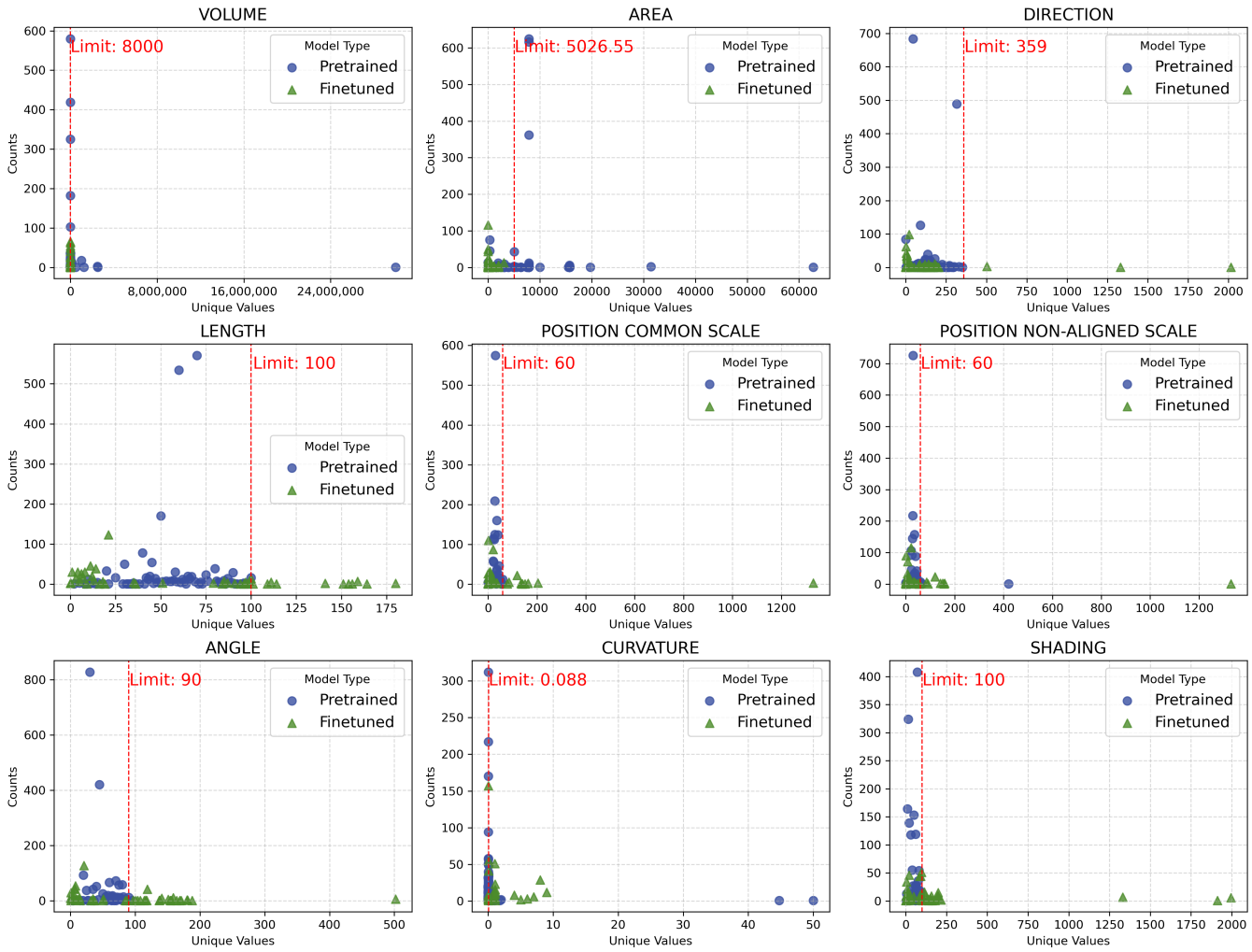

Figure 2: Distribution of unique values for different perceptual tasks comparing between pretrained and finetuned models. The red lines show the maximum allowed value for each task, with pretrained models often exceeding these limits while finetuned models stay within bounds.

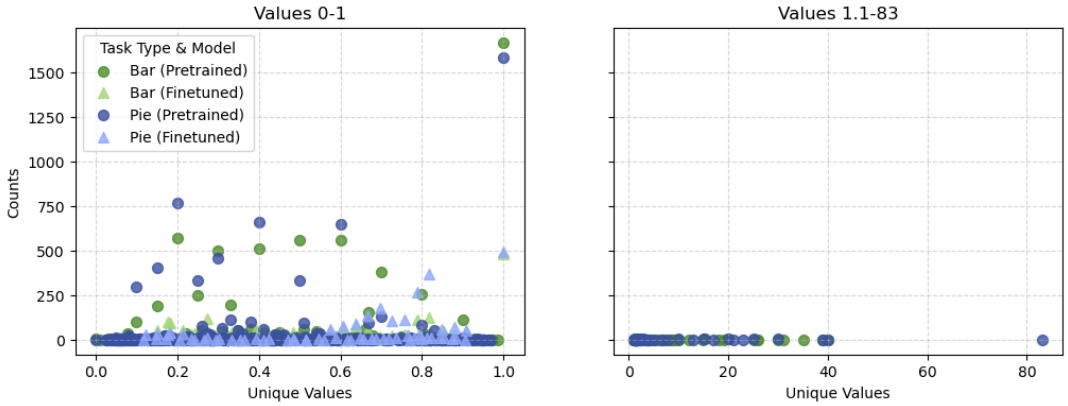

Figure 3: Distribution of unique values for bar and pie chart outputs, split between values in ranges 0-1 and 1.1-83. The left plot shows most responses focused in the expected 0-1 range, while the right plot reveals some outlier predictions extending up to 83.

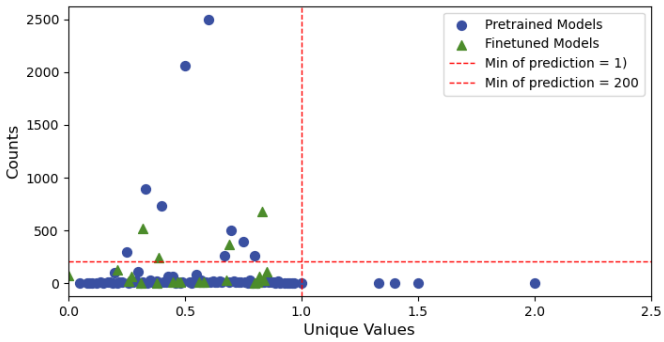

Figure 4: Comparison of unique value counts between pretrained and finetuned models across all five perceptual tasks (Type1-Type5). Pretrained models show higher count spikes and some values beyond 1.0, while finetuned models maintain lower counts and generally stay within the expected range.

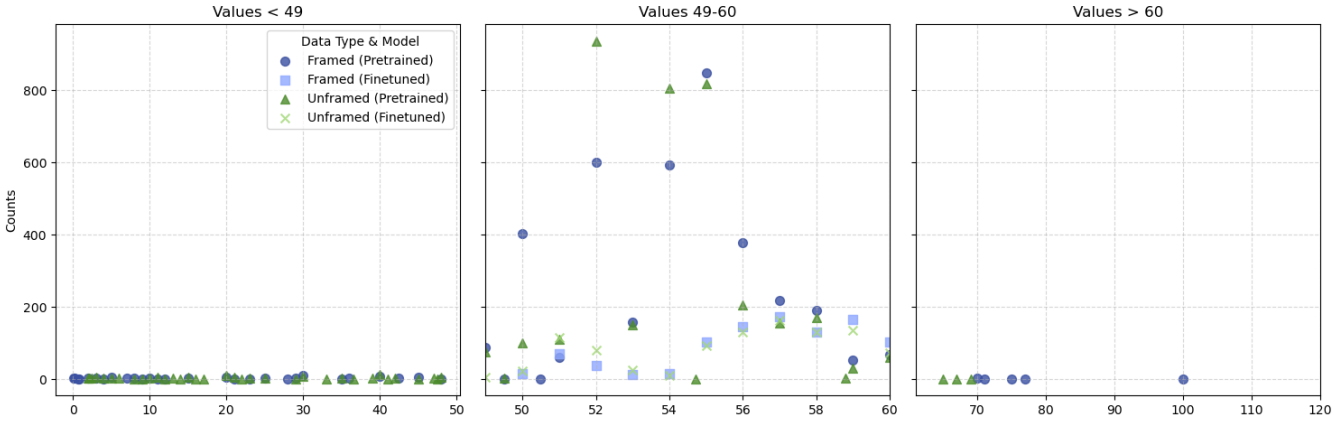

Figure 5: Distribution of values across three ranges (below 49, 49-60, and above 60) comparing framed and unframed tasks for both pretrained and finetuned models. While most values concentrate in the middle range (49-60), both pretrained model types occasionally produce values beyond 60.

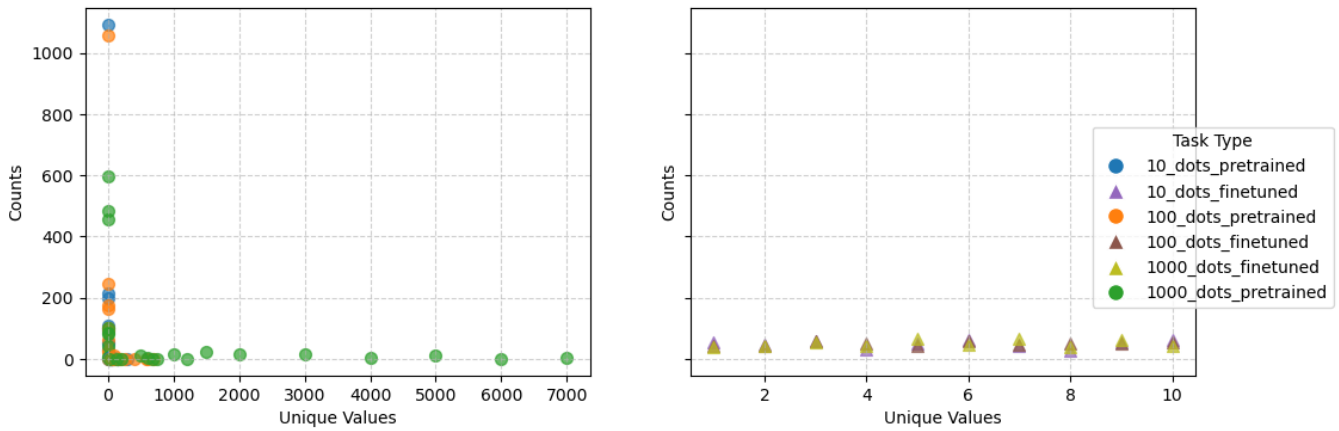

Figure 6: Distribution of unique values for tasks with varying numbers of dots (10, 100, and 1000), comparing pretrained and finetuned models.