What Matters when Building Vision Language Models for Product Image Analysis?

Ameni Trabelsi* Maria Zontak* Yiming Qian* Brian Jackson Suleiman Khan Umit Batur
Amazon Inc.
Seattle, USA

Abstract

This paper investigates multi-modal large language models (MLLMs) for predicting product features from images, comparing fine-tuned versus proprietary models. We introduce two domain-specific benchmarks: (1) Inductive Bias vs. Image Evidence (IBIE) Benchmark, which evaluates MLLMs' ability to distinguish between image-derived features and latent knowledge, and (2) Catalog-bench, which assesses feature prediction using Catalog terminology. Our fine-tuned model outperforms proprietary models like Gemini by 9.4% and 29.13% on these benchmarks respectively. We address the crucial aspect of computational efficiency, exploring cost effective deployment solutions under limited hardware resources. The significance of this work extends beyond e- commerce to physical retail, where efficient MLLMs are essential for real-time processing of visual data from store cameras and shelf sensors. These models enable automated inventory management, produce quality monitoring, and planogram compliance while operating within in-store computing constraints. This capability is particularly valuable for physical retail environments where immediate decisions about restocking and quality control are critical, while also enabling real-time assistance to customers seeking information about product details, ingredients, and nutritional content.

1. Introduction

Vision Language Models (VLMs) or Multimodal Large Language Models (MLLMs), designed to bridge the gap between visual and textual understanding, can analyze images and videos in conjunction with text, enabling a wide range of applications. For example, MLLMs can generate descriptive captions for images, answer questions about visual content, and even follow complex instructions involving both visual and textual cues. However, the question remains how well these general MLLMs perform in specialized domains, in other words: *Are proprietary models the right choice for domain specific applications?* Recent advancements in generative AI and specifically LLMs, have paved the way for innovative approaches to transform variety of industrial applications. Specialized LLMs are developed to safely answer medical question (e.g., Med-PaLM 2 [10]), generate code (e.g., GitHub Copilot [28]), personalize tutoring, automate speech recognition and translation and much more. Retail product analysis, an inherently multimodal domain, offers an exciting playground for testing and utilizing MLLMs capabilities.

In this paper, we investigate the key design choices in developing efficient multi-modal large language models (MLLMs) for product image analysis. To explore this, we focus on the generation of product attributes from a single image. We first investigate whether existing proprietary models, like Google Gemini and ChatGPT, sufficiently address this task. To guide our analysis, we evaluate both open-source and proprietary models using domain-specific benchmarks that necessitate specialized product knowledge and nuanced understanding. We further delve into techniques for fine-tuning MLLMs that preserve their general capabilities, thereby mitigating the risk of overfitting. Our findings reveal that fine-tuning a model, even when compared to state-of-the-art proprietary models, can lead to significant performance improvements in product-specific domain. While the authors of [43] introduced a novel training methodology, optimizing nuanced attribute extraction from product images [42], our work takes a broader perspective. Inspired by [15], who explored efficient ways to leverage smaller language models for domain-specific applications, we examine the development and deployment of MLLMs tailored for attribute extraction from product images. The contributions of this paper are:

- We introduce two benchmarks, Inductive Bias vs. Image Evidence (IBIE) Benchmark and Catalog-bench, carefully curated from product pages with high-quality labels.

- We affirm that fine-tuning open-source models on domain-specific datasets can surpass the performance of proprietary models on domain-specific tasks.

*Authors contributed equally

- We emphasize the importance of combination of in-domain dataset, reasoning-based training and generic data to preserve the model's general capabilities.
- We investigate cost-effective deployment and analyze the trade-offs between cost and performance, essential for successfully deploying MLLMs in physical retail applications due to the strict computational requirements of industrial production settings.

2. Related Literature

2.1. Multimodal Large Language Models (MLLMs)

While Vision Language Models have a rich history, Flamingo [3] and BLIP-2 [16] have laid the foundation for the recent surge in MLLMs development. These works pioneered the bridging of the representational gap between visual encoders (such as OpenAIClip [34] or SigLip [40]) and LLMs. This was achieved through an adapter mechanism that transforms visual features into visual tokens, subsequently processed autoregressively by the LLM to generate the desired textual output. Specifically, Flamingo introduced gated cross-attention layers that were interleaved with frozen language layers, enabling selective integration of visual information into text generation. BLIP-2 further simplified Vision-Language alignment by introducing a Q-former, the only trainable part of its architecture, based on a lightweight, 12-layer transformer encoder. Q-former that sits between the image encoder and the language model undergoes a two-stage training. In the first stage, the Q-former learns (using trainable queries) to extract image features most relevant to corresponding text, connecting to a frozen image encoder. In the second stage, it is trained to generate output features that can be interpreted by the LLM to produce corresponding text. BLIP- 2 achieved superior performance while using 54 times fewer trainable parameters than Flamingo-80B.

The adapter paradigm has since become the cornerstone for many subsequent MLLMs, with LLaVA [24, 25] taking an architectural efficiency one step further, introducing a lightweight, 2-layer MLP adapter for mapping visual embeddings to language model token embeddings. This efficient approach required only 595K image-text pairs for initial alignment, compared to the substantially larger 100 million samples needed in BLIP-2 pretraining. Subsequent works like MM1 [27] and Idefics2 [14] further explored the optimal adaptation approach, each reaching a different conclusion, nevertheless emphasizing the trade-off between the number of visual tokens and final performance.

Beyond architectural choices for the adapter, these and other works have emphasized the critical role of well-curated datasets. Flamingo pioneered a web-driven, interleaved dataset that combined multiple images and text within each sample. BLIP-2 leveraged caption bootstrapping, a technique whose effectiveness was demonstrated by its predecessor [17]. LLaVA incorporated academic task-oriented data and improved visual instruction tuning [24, 26]. VILA [19] adopted a similar approach to LLaVA for visual-language alignment but further expanded its in-context learning capabilities by utilizing an interleaved dataset [41] during pretraining. MM1 paper further analyzed the ideal ratios of interleaved, image-caption and text datasets for different stages of MLLM training.

Models like Idefics2 and Kosmos2 [33] explored the need for large-scale datasets and bounding boxes to enhance grounding abilities. Recently, Cambrian-1 [38] managed to significantly reduce the number of visual tokens while maintaining competitive performance. This was accomplished through a novel dynamic and spatially-aware connector integrating visual features from multiple models, as well as Cambrian-7M, a large-scale, high-quality visual instruction-tuning dataset. Cambrian-7M was curated from the larger Cambrian-10M dataset and outperformed it on a variety of benchmarks after instruction tuning, highlighting the importance of well curated datasets. Our work further highlights the importance of a well-curate dataset, especially ensuring training data closely matches the target task for optimal model performance.

While the open-source landscape of MLLMs is flourishing, the state-of-the-art performance is mostly governed by numerous proprietary models, such as Google's Gemini, Open-AI GPT-4o, Anthropic Claude. While these models might be extremely appealing due to their overall strong performance and ease of usage, the closed nature of these models limits the ability to adapt those models to specific domains, which require more specialization.

2.2. MLLMs for Retail Applications

There are multiple applications of AI and LLMs in retail. LLMs, acting as chatbots, automate customer service and shopping assistance [5], providing faster responses and personalized product recommendations. This not only improves the customer experience but also accelerates sales and frees up human resources. Recent MLLMs that include visual signals (images or videos) can be utilized in domains in which visual features enhance the quality of the AI generated result, such as search based on visual features and generative listings from product images [4].

While previous studies [7,8,32] focused on text-based attribute extraction, our approach uniquely uses image data to generate visual attributes (such as brand, color, style, shape, and specific product features, as illustrated in Table 1 and Fig. 1 a). While earlier work [20] also recognized the importance of using visual cues for product attribute generation, their reliance on OCR and object detectors sets them apart. In contrast, we propose a streamlined MLLM solution for this task.

Recent academic work [43] tackled the problem of attribute extraction from a product image with a learning

Example 1:
Question: Concentrating on the product in the image, determine if you can infer the following "Resealable bag" from the image?

Human answer: Yes

Answer with reasoning: Yes, the image clearly shows a resealable plastic bag containing sliced almonds.

Example 2:
Question: Concentrating on the product in the image, determine if you can infer the following "Good source of fiber" from the image?

Human answer: No

Answer with reasoning: Yes, almonds are a good source of dietary fiber, so the image of sliced almonds from Fisher's Chef's Naturals indicates that this product would be a good source of fiber.

Example 3:
Question: Concentrating on the product in the image, determine if you can infer the following "Good source of fiber" from the information *physically present in* the image?
Human answer: No

Answer with reasoning: No, the package does not make any claims about the almonds being a good source of fiber or provide nutrition facts showing the fiber content. While almonds are known to contain fiber, this specific fact cannot be confirmed from the information given on the package in the image alone.

TABLE 1 **Model Inference from Latent Knowledge vs. Information in the Image**: In Example 1 proprietary model is reasoning based on the information it can recognize it in the image. However, in Example 2 the model relies on its latent knowledge. Adding "*physically present in*" to the prompt of Example 2 helped the model to reason based on the image information (Example 3).

by comparison strategy, utilizing a model's judgment to compare two sample products and predict the correct attribute values given the provided images. Authors of [39] proposed a general multiple instance visual component to bridge the gap between various image inputs with off-the-shelf MLLMs and demonstrated its effectiveness on product attribute prediction from multiple images. Differently from these works, we focus on an efficient approach for attribute generation that optimizes inference latency by using a single sample product to infer attribute value.

3. Domain–Specific Benchmarks for Model Choice

When deciding whether to utilize a proprietary model or fine-tune an open-source model, the development of domain-specific benchmarks is crucial. These benchmarks should focus on tasks unique to product understanding, complementing existing generic benchmarks, and serving as key indicators of model effectiveness within the specific domain. For instance, while text understanding from product images is important, creating an OCR question-answering benchmark for product images is unnecessary, as existing benchmarks like TextVQA and MMMU already cover OCR and chart understanding. Instead, the focus should be on creating product-understanding benchmarks that assess an MLLM's ability to interpret relevant information from product images. To address this, we develop two novel benchmarks: the Inductive Bias vs. Image Evidence (IBIE) Benchmark, which evaluates the MLLM's ability to reliably reason about product information from images, and Catalog-Bench that assesses model understanding of domain-specific terminology. These benchmarks provide a more comprehensive and targeted evaluation framework for MLLMs in the context of product image analysis.

3.1. Inductive Bias vs. Image Evidence (IBIE) Benchmark

The development of specialized MLLMs often uses web data, including product pages with images and text descriptions. However, product images and text may not perfectly align, as text often mentions features not visible in images. This misalignment can cause MLLMs to "hallucinate" or generate incorrect information, as noted in [6]. To evaluate MLLMs' ability to distinguish between visible image features versus learned knowledge, researchers created the IBIE benchmark. This benchmark was developed by having data associates verify if specific features mentioned in product descriptions could be directly inferred from product images. The final IBIE benchmark contains 1,214 questions based on 209 images covering 101 product types.

Table 1 illustrates how IBIE benchmark works. The first example illustrates a case where the feature <Resealable bag> can be inferred directly from the visual content (the blue arrow at the top of the bag shows that the bag can be resealed). The second example shows that sometimes the model will rely on inductive bias and conclude that almonds are <Good source of fiber>, despite that this feature

is not mentioned directly on the package. While generally it makes sense to conclude that the almonds are a good source of fiber, this information cannot be directly inferred from the image, thus it is a form of hallucination. In order to avoid such hallucinations, our benchmark can guide prompt optimization. For example, we can further modify the prompt to guide the model to rely on the *information physically present in* the image, as shown in the third example. Using this prompt helps the proprietary model to correctly identify that <Good source of fiber> feature cannot be inferred from the image.

Several recent works suggested that Chain-of-Thought reasoning can help improve model performance and avoid hallucinations, we therefore evaluate MLLM accuracy on IBIE benchmark with and without reasoning. In the first case we ask the model to provide first the reason for the answer and then Yes / No response. In the second case, only Yes/ No response is required.

3.2. Catalog-Bench

To ensure the model's ability to extract attributes using standardized terminology for downstream applications like search and visual question answering, we have developed a comprehensive benchmark of multiple-choice questions [1]. This benchmark with a total of 440K questions encompasses 53 different tasks related to 44 unique product attributes and 66 product types. The number of labels varies across tasks, as shown in Fig. 1 a., with some tasks like item shape and pattern having over 30 labels. Figure 1b. illustrates several test images for an example task of predicting dress occasion type, demonstrating the challenge of this task for models without the specialized knowledge.

4. Design Choices for Domain-Specific MLLM

In this section we address two topics. First, we assess the performance of proprietary models on the task of product feature prediction from product images. We benchmark Gemini and GPT4 on the two product-specific benchmarks we described in Sec. 3 and compare those to our in-house fine-tuned LLaVA model. Second, we describe our design choices of the in-house fine-tuned LLaVA.

4.1. Proprietary vs. Fine-tuned Models

While proprietary models, like Gemini and GPT4, excel in general multi-modal tasks, their effectiveness may vary for specific applications. Our domain-specific benchmark comparisons between proprietary and fine-tuned open-source models, presented in Table 2, demonstrate that task-specific fine-tuning on well-curated datasets can achieve superior performance[2].

[1]We thank Suren Kumar and Sayantan Marik for their support with Catalog-bench

[2]Due to legal constrains, the performance numbers are reported relatively to a **random** number we chose as a baseline.

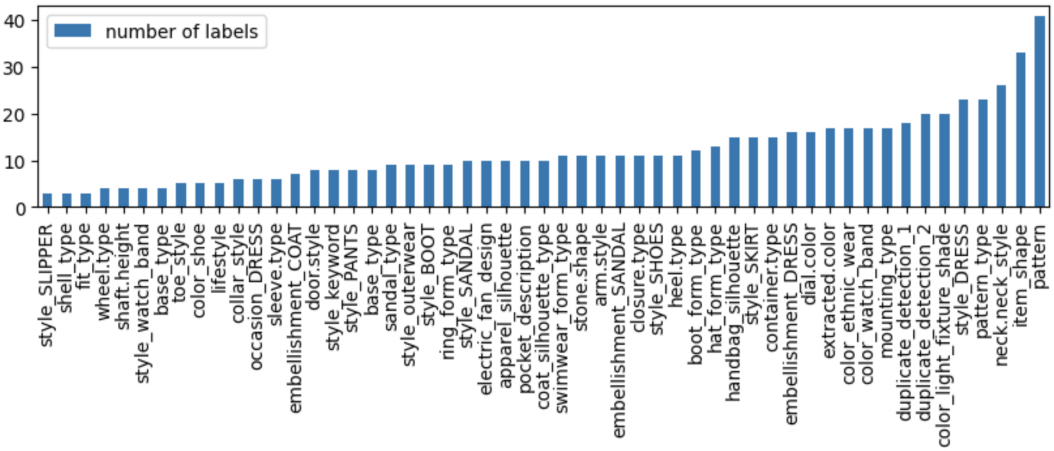
Results on IBIE Benchmark: As expected, Gemini performs better than open-source LLaVA, but it performs worse than fine-tuned LLaVA (+20.02% vs. +29.4%). Interestingly, IBIE with reasoning yields consistently worse performance compared to simple Yes/ No answers for all MLLMs we evaluated (+16.72% vs. +21.09% for Gemini). This is expected for models that were finetuned on related tasks, however it is surprising in the zero-shot performance and should be further investigated. To conclude, **the IBIE benchmark highlights that even powerful MLLMs show limited proficiency in leveraging visual evidence to assess image-text alignment. However, this ability can be significantly enhanced through targeted fine-tuning**.

Results on Catalog-bench: The MLLMs performance on this benchmark correlate with their general abilities, specifically GPT4 Turbo leads on Google Gemini on average performance by 6.38%. Finetuning a relatively small (see Sec. 4.2) open-source model yields much better performance, especially for domain specific attributes. Specifically color, which is inherent to image understanding, benefits less from finetuning (+32.66% for GPT4 vs. +39.95% for finetuned LLaVA) compared to dress style (-5.48% for GPT4 vs. +20.38% for finetuned LLaVA). Furthermore, we see that for attributes with more possible label options, like dress style with 23 labels, the performance of proprietary models degrade (with -14.58% and -5.48% for Gemini and GPT4 respectively), compared to dress occasion attribute with 6 labels only (+6% and +27.33% for Gemini and GPT4 respectively). To conclude, **zero-shot performance of proprietary models on retail-specific terminology is promising, but further fine-tuning can considerably boost performance**. In-context learning based on few-shot prompting can boost the performance of proprietary models. Our experiments with 4-shot inference using Gemini showed an improvement of +12% in the average performance across all tasks of the Catalog-bench, however this still does not close the gap to the fine-tuned model.

4.2. In-house Fine-tuned Model Details

Our fine-tuned model (the winning model from Table 2) follows LLaVA model architecture [24]. Next we elaborate on the choice of vision encoder and LLM. When selecting an LLM, while larger models generally perform better for both language and multimodal tasks, practical considerations must be balanced. Although 80B parameter models may offer superior performance, they present challenges in terms of: higher computational costs, slower iteration cycles, greater environmental impact and increased inference latency. For specific applications, medium-sized models can be more practical, offering better efficiency and cost-effectiveness while maintaining adequate performance.

Apple's MM1 [27] and HuggingFace's Idefics2 [14] has recently affirmed that smaller (7-8B parameter) models can achieve competitive performance compared to much larger



(A) Distribution of number of labels per task

(B) Example of test images for Dress Occasion task "*By examining the dress pictured, categorize which occasion is the dress designed for based on styling cues? Labels: A.Casual B. Club C. Cocktail D. Evening Gown E. Wedding F. Work* "

FIGURE 1 **Catalog-Bench Benchmark**

Benchmark	Details	Baseline	Gemini Pro	GPT4 Turbo	Open-source LLaVA	**Our Fine-Tuned LLaVA**
IBIE	with reason	+0%	+16.72%	-	+ 8.3%	**+21.09%**
	no reason	+0%	+20.02%	-	+12.29%	**+29.4%**
Catalog-bench	Average over 53 tasks	+0%	+5.09%	+11.47%	-5.47%	**+34.22%**
	Color Task (17 labels)	+0%	+35.62%	+32.66%	+30.82%	**+39.95%**
	Dress Occasion Task (6 labels)	+0%	+6%	+27.33%	+12.76%	**+37.58%**
	Dress Style Task (23 labels)	+0%	-14.58%	-5.48%	-29.22%	**+20.38%**

TABLE 2 **Proprietary vs. Fine-tuned Open-Source MLLMs performance on product domain-specific benchmarks.** For proprietary models, prompts were optimized to produce the best results. For both benchmarks, performance is reported as a differences with respect to a random baseline (a randomly chosen number), due to a restriction on publishing in-house absolute performance.

models. Phi3-vision-instruct model [1], took it one step further demonstrating state-of-the-art results on a broad set of vision benchmarks (including MMMU and Text-VQA), surpassing bigger models such Llama3-Llava-Next-8B. While the smaller size of Phi3 (less than 4B parameters) is appealing, our experiments with this model showed its tendency to hallucinations, which also aligns with the inferior POPE metrics (average F1 score of 85.8% as reported in [29] vs. average scores of 86.17% and above for different variants of LLaVA1.5-Mistral7B [30] in Table 3). We hypothesize that one possible reason is the scale of the dataset (500B vision and text tokens) that instruct-Phi3 used to become competitive with bigger models. Big datasets might pose a quality problem (poor alignment of text and images), and might encourage the model to hallucinate. Therefore, **we utilize Mistral 7B as the base model for our experiments**.

Regarding vision encoder selection, while OpenAI CLIP was traditionally preferred, recent models like TinyLLaVA, Instruct-Phi3, and Cambrian have adopted SigLIP-SO400M [40]. This 400M-parameter Shape Optimized encoder offers higher-resolution image features (729 vs. CLIP's 576) and was trained on the more extensive We-

		TextVQA	GQA	MMMU	POPE (hallucination benchmark)			
LLaVA Version	Visual Encoder			val	Random	Popular	Adversarial	Average
1.5	Frozen SigLip	60.92%	63.48%	36.6%	88.25%	87.04%	**85.38%**	**86.89%**
1.5	Frozen OpenAI Clip	57.39%	62.25%	36.3%	**88.62%**	**86.65%**	84.24%	86.50%
Next	Finetuned SigLip	**69.12%**	**63.79%**	**40.3%**	87.38%	86.23%	84.92%	86.17%
Next	Finetuned OpenAI Clip	69%	63.76%	36.3%	88.25%	86.64%	85.1%	86.66%

TABLE 3 **Choosing Vision Encoder: SigLip vs. OpenAI Clip within LLaVA framework**. See Sec. 4.2 text for details.

bLi dataset (10B images) compared to CLIP's 400M dataset. In Table 3, we present a comparison of SigLip and OpenAI Clip, considering both frozen and finetuned vision encoder (LLaVA1.5 [24] and LLaVA-Next [26] respectively). LLaVA1.5 embeds a resized input image with frozen vision encoder, while LLaVA-Next accommodates images of various resolutions by breaking the image into grids, which are embedded and concatenated to get a richer representation. To optimize the multi-patch feature embedding, LLaVA-Next finetunes the vision encoder during the finetuning stage with a bigger corpus of an improved visual instruction tuning data mixture for OCR and Chart understanding tasks [13,24]. While SigLip generally outperforms OpenAI Clip on most metrics without finetuning, we find that finetuned OpenAI Clip achieves comparable performance to SigLip, with the TextVQA difference narrowing significantly from 3.53% to 0.12%, and a superior Pope score. These results suggest that finetuning vision encoder can effectively minimize performance gaps between vision encoders. However, it is crucial to recognize that the original benefits of the vision encoder remain significant in the absence of finetuning. Although the increased number of image tokens in LLaVA-Next leads to improved performance on OCR tasks, it also results in a slower inference speed of 2.3 images/sec compared to 6.5 images/sec, thus requiring more computations resources to process the same number of images. In our experiments, **we prioritized efficient inference and therefore opted to utilize the frozen SigLip encoder within the LLaVA1.5 framework**.

Finally, we trained LLaVA1.5 [24] with Mistral7B and SigLip using the original 2-stage approach. The pretraining stage utilized the data from [21], including Google's Conceptual Captions Dataset [31, 35]. The fine-tuning stage dataset is detailed in Section 4.3. We followed the training parameters recommended in the code repository [23] with a few adjustments. First, a slower learning rate of $5e{-}6$ produced smoother training loss during Mistral7B fine-tuning. Second, despite using 80GB GPUs (A100 or H100), we had to reduce the batch size to 8 samples and increased accumulation steps to 2 to maintain the overall sample count. The pretraining took up to 10 hours (on 8 40GB A100 GPUs). The finetuning on our biggest data corpus took up to 24 hours (depending on the GPU type) with 8 GPUs.

Data	Percentage
General VQA (Similar to LLaVA)	40%
Multiple-Choice Questions (Catalog-bench)	20%
Binary Questions (IBIE)	20%
Open-Generation Questions	20%

TABLE 4 **In-house Fine-tuning Dataset Composition**

4.3. Task-Specific Training Data Ablation Study

While proprietary models are often considered capable of complex reasoning due to their ability to handle task-related instructions [2, 37], we investigate whether fine-tuning compromises this general reasoning ability. We conduct an ablation study on the composition of the training dataset used for product feature value prediction to assess the impact of fine-tuning on a model's broader capabilities.

Table 4 summarizes our finetuning data, which includes:

General VQA: We utilize the finetuning data of LLaVA [22], except for the GPT assisted visual instruction data. Instead, we follow similar generation strategy and prompt an in house LLM to curate instruction following data involving visual content covering three main tasks: multi turn conversations, complex reasoning and detailed image captioning. We use images and image labels (captions and objects bounding boxes) from the Coco dataset.

Multiple Choice Questions for Attribute Prediction: Given a product image, select the correct label from a set of options of the task (the training split of the Catalog-bench).

Binary Questions: Given a product image and a feature, determine if the feature can be inferred from the information evident in the image (the training split of the IBIE benchmark).

Open-Generation Questions: Given a product image generate a value of the given attribute. We used curated image-attribute pairs from high-quality datasets like Catalog-bench training split, and additional catalog data, which was manually curated to have high alignment between visual and textual information. We augment some of the training samples to follow the chain-of-thought (CoT) reasoning where the target prediction follows a CoT reasoning about the answer. The goal is to avoid overfitting to the task by inciting the model to maintain its reasoning capabilities while learning the task at hand. In addition,

Training Data	TextVQA	GQA	Catalog-bench	IBIE
General VQA (baseline)	**+0%**	+0%	+0%	+0%
Multi.-Choice + Binary-Quest. + Open-Gen.	-28.98%	-36.13%	+39.16%	+18.71%
General VQA + Multi.-Choice	-1.32%	+1.33%	**+39.72%**	+3.3%
General VQA + Multi.-Choice + Open-Gen.	-0.09%	+1.78%	+39.57%	+4.18%
General VQA + Multi.-Choice + Binary-Quest. + Open-Gen.	-0.39%	+1.98%	+39.65%	+19.53%
General VQA + Multi.-Choice + Binary-Quest. + Open-Gen. + **Spatial Grounding**	-0.01%	**+2.26%**	+39.69%	**+20.41%**

TABLE 5 **Finetuned models performance on public benchmarks as well as Catalog-bench and IBIE benchmarks.**

FIGURE 2 **Samples from the Spatial Grounding Dataset**

similarly to [32], we add negative training samples where the feature in the input query cannot be found in the image and annotate these samples with the label "Not Obtainable". This data helps to reduce hallucinations by falling to "Not Obtainable" when the model is not confident.

Spatial Grounding Questions: We further developed a specialized spatial grounding dataset to enhance the model's localization capabilities in product understanding. The dataset consists of 25,000 products across 26 categories, where each product image is paired with attribute descriptions and their precise spatial coordinates, as illustrated in Fig. 2. Human annotators manually identified specific pixel coordinates (x, y) corresponding to each attribute description. This dataset serves two key purposes: 1) improves the model's visual grounding ability (as illustrated in the Supplementary material); and 2) enhances attribute prediction accuracy while reducing hallucination artifacts.

To enhance the model's ability to associate textual descriptions with specific locations in images and vice versa., we designed two types of conversational training tasks:

1. **Coordinate Identification**: Given an image of size 384×384 pixels, locate the pixel coordinates corresponding to the feature described as "{attribute description}". **Answer**: (x, y)

2. **Feature Recognition**: In a 384×384 pixel image, identify the product feature located near coordinate (x, y). Provide only the feature name. **Answer**: {attribute description}

In Table 5, we assess the generic capability using two public benchmarks TextVQA [36] and GQA [11]. We also assess the performance of the model on the target tasks using our evaluation datasets (Catalog-bench and IBIE). First, we compare the baseline model trained only on General VQA against the finetuned model on product feature prediction where only task specific dataset are used (no General VQA). We notice that while the target task metrics improve considerably (+39.16% and +18.71% on Catalog-bench and IBIE respectively), both TextVQA and GQA considerably degrade when removing the General VQA dataset. This shows that unless the training dataset includes General VQA, the finetuned model will degrade in performance in public General QA benchmarks. Furthermore, we experiment with combining the General VQA and the task specific datasets. We experiment by adding the Multiple-Choice dataset in one experiment, both Multiple-Choice and Open-Generation datasets in another, and all three task specific datasets in a third experiment. We recall that Open-Generation is augmented with CoT reasoning. First, we notice that the performance on the public benchmark has been mostly recovered compared to the previous finetuned model with TextVQA increase from -28.98% to -1.32% and GQA from -36.13% to -1.33% when adding the General VQA compared to baseline. Second, we see that the addition of Multiple-Choice and Open-Generation have slightly improved the performance on IBIE (3.3% and 4.18% respectively). However, we see the most boost (+19.53%) after including training data, which tackles the same task being evaluated in IBIE (aka Binary Questions). This shows that while training on similar tasks (like multiple choice feature value prediction or open generation feature value prediction) helps improve the performance on the IBIE benchmark, finetuning on the specific task is still necessary to achieve the best performance. Furthermore, it is worth highlighting that adding Open-Generation dataset in the training only helped slightly the General benchmarks. This shows that adding CoT reasoning in training dataset has only minimal improvement on the generic aspect of the finetuned model. Finally, we see that further **challenging the model with spatial grounding task, improves the model across both generic and domain-specific tasks**, yielding the best performance on GQA (+2.26%) and IBIE (+20.41%) benchmark. This illustrates that **spatial grounding training task helps the model to reason better and hallucinate less**.

In conclusion, this experiment demonstrates that

Method	IBIE	Catalog-bench
Full Precision (baseline)	+0%	+0%
Gemini	-8.93%	-29.58%
4bit Precision	-11.76%	-21.54%
GPTQ	-8.32%	-16.88%
AWQ	**-8.17%**	**-16.01%**
SqueezeLLM	-9.28%	-17.98%

TABLE 6 **Comparison of the performance of Post Training Quantization Approaches vs. Full Precision Model and Google Gemini on IBIE and Catalog-bench.** AWQ shows the least performance decrease compared to its counterpart. 4bit precision model has lower performance compared to post training quantization. All quantized models still outperform Gemini on IBIE and Catalog-bench.

achieving diversified model fine-tuning is possible when the tasks complement each other. However, **for the model to perform well on the actual inference task, including General QA, it is crucial to include relevant training data that closely resembles the task's requirements**. Simply diversifying the fine-tuning tasks without aligning them with the target inference task may not guarantee optimal performance.

5. Cost-Effective Deployment Trade-Offs

Deploying MLLMs in real applications is still challenging due to the strict computational requirements of the industrial production settings. For instance, achieving real time inference of LLaVA models on restricted GPU memory would require applying optimization techniques like GPTQ [9], SqueezeLLM [12], or Activation-aware Weight Quantization (AWQ) [18]. In this section, we look into the effect of model optimization techniques on the model performance for the case of product feature prediction. More specifically, we look into 4 bit quantization. We compare the performance gap between loading the model with 4 bit precision and post-training quantization techniques. These experiments focus on quantized 7B Mistral based LLaVA-like models with single patch as they can be deployed with consumer-grade GPUs (e.g., 24GB memory).

In Table 6, we report the performance of the quantized variants of the finetuned model, and compare it to the full precision model as well as Google Gemini. First, all quantized variants downperform the full precision model in both datasets. On the binary classification task (IBIE), GPTQ reduces the accuracy by 8.32%, while AWQ reduces it by 8.17%. Furthermore, we see a clear advantage of using post training quantization compared to directly using 4bit precision. Quantization methods like AWQ helped reduce the performance drop from -11.76% to -8.17% on IBIE. In both datasets, the AWQ outperformed its counterpart, followed by GPTQ and SqueezeLLM. It is also worth noting that though the studied optimization reduces the performance on both tasks, they perform on par with zero-shot Gemini on IBIE. Such a solution could prove beneficial for companies constrained by the use of open-source models and limited hardware resources. In future work, we will explore other optimization techniques like continuous batching and batch-specific memory optimizations which have less effect on model performance during inference.

6. Discussion and Future Work

In this paper, we investigate key factors that influence the development of effective task-specific Multimodal Large Language Models (MLLMs) for specialized applications, focusing particularly on product attribute extraction from images. Our research demonstrates that open-source models, when finetuned on carefully curated domain-specific datasets, can surpass the performance of proprietary state-of-the-art models such as Google Gemini and GPT-4 Turbo in specialized tasks.

To rigorously evaluate model performance, we developed two comprehensive benchmarks: 1) The Inductive Bias vs. Image Evidence (IBIE) benchmark, which assesses a model's capability to accurately infer product features from visual information; and 2) Catalog-Bench, a multiple-choice question benchmark that evaluates a model's ability to extract product attributes from predefined value sets given an input image. Through comprehensive ablation studies, we demonstrate that dataset quality and composition are critical for model performance. Our findings reveal that successful model deployment requires training data that closely aligns with the target task requirements, including general question-answering capabilities. Furthermore, our preliminary experiments suggests that for complex tasks, few-shot learning approaches yield suboptimal results compared to proper fine-tuning. Finally, we tackle the resource limitations in real applications by assessing the performance gap when using optimization techniques to reduce the models inference computational cost. We report a gap of at least 8.17% on IBIE and 16.01% on Catalog-bench when using AWQ 4bit Post Training Quantization.

While Table 5 shows our fine-tuned model substantially outperforms proprietary models on domain-specific tasks, we acknowledge performance gaps in generic capabilities, particularly in OCR understanding. For instance, our preliminary analysis on the TextVQA benchmark revealed that ChatGPT achieves 28% improvement over the baseline, significantly outperforming our fine-tuned LLaVA model, which showed only an 11% improvement. Improvement of OCR capabilities is the focus of our curent work.

While we experimented with attribute extraction from product images, we believe that our findings are relevant for any retail application that requires domain-specific knowledge.

References

[1] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Qin Cai, Martin Cai, Caio César Teodoro Mendes, Weizhu Chen, Vishrav Chaudhary, Dong Chen, Dongdong Chen, Yen-Chun Chen, Yi-Ling Chen, Parul Chopra, Xiyang Dai, Allie Del Giorno, Gustavo de Rosa, Matthew Dixon, Ronen Eldan, Victor Fragoso, Dan Iter, Mei Gao, Min Gao, Jianfeng Gao, Amit Garg, Abhishek Goswami, Suriya Gunasekar, Emman Haider, Junheng Hao, Russell J. Hewett, Jamie Huynh, Mojan Javaheripi, Xin Jin, Piero Kauffmann, Nikos Karampatziakis, Dongwoo Kim, Mahoud Khademi, Lev Kurilenko, James R. Lee, Yin Tat Lee, Yuanzhi Li, Yunsheng Li, Chen Liang, Lars Liden, Ce Liu, Mengchen Liu, Weishung Liu, Eric Lin, Zeqi Lin, Chong Luo, Piyush Madan, Matt Mazzola, Arindam Mitra, Hardik Modi, Anh Nguyen, Brandon Norick, Barun Patra, Daniel Perez-Becker, Thomas Portet, Reid Pryzant, Heyang Qin, Marko Radmilac, Corby Rosset, Sambudha Roy, Olatunji Ruwase, Olli Saarikivi, Amin Saied, Adil Salim, Michael Santacroce, Shital Shah, Ning Shang, Hiteshi Sharma, Swadheen Shukla, Xia Song, Masahiro Tanaka, Andrea Tupini, Xin Wang, Lijuan Wang, Chunyu Wang, Yu Wang, Rachel Ward, Guanhua Wang, Philipp Witte, Haiping Wu, Michael Wyatt, Bin Xiao, Can Xu, Jiahang Xu, Weijian Xu, Sonali Yadav, Fan Yang, Jianwei Yang, Ziyi Yang, Yifan Yang, Donghan Yu, Lu Yuan, Chengruidong Zhang, Cyril Zhang, Jianwen Zhang, Li Lyna Zhang, Yi Zhang, Yue Zhang, Yunan Zhang, and Xiren Zhou. Phi-3 technical report: A highly capable language model locally on your phone, 2024. 5

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 6

[3] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikoł aj Bińkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 23716–23736. Curran Associates, Inc., 2022. 2

[4] Amazon. Generative Listings. Accessed on July 14th, 2024. 2

[5] Amazon. Rufus, 2024. 2

[6] Elmira Amirloo, Jean-Philippe Fauconnier, Christoph Roesmann, Christian Kerl, Rinu Boney, Yusu Qian, Zirui Wang, Afshin Dehghan, Yinfei Yang, Zhe Gan, and Peter Grasch. Understanding alignment in multimodal llms: A comprehensive study, 2024. 3

[7] Ansel Blume, Nasser Zalmout, Heng Ji, and Xian Li. Generative models for product attribute extraction. In Mingxuan Wang and Imed Zitouni, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 575–585, Singapore, Dec. 2023. Association for Computational Linguistics. 2

[8] Chenhao Fang, Xiaohan Li, Zezhong Fan, Jianpeng Xu, Kaushiki Nag, Evren Körpeoglu, Sushant Kumar, and Kannan Achan. Llm-ensemble: Optimal large language model ensemble method for e-commerce product attribute value extraction. *ArXiv*, abs/2403.00863, 2024. 2

[9] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022. 8

[10] Google. Med-PaLM. Accessed on July 14th, 2024. 1

[11] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019. 7

[12] Sehoon Kim, Coleman Hooper, Amir Gholami, Zhen Dong, Xiuyu Li, Sheng Shen, Michael W Mahoney, and Kurt Keutzer. Squeezellm: Dense-and-sparse quantization. *arXiv preprint arXiv:2306.07629*, 2023. 8

[13] Hugo Laurençon, Lucile Saulnier, Léo Tronchon, Stas Bekman, Amanpreet Singh, Anton Lozhkov, Thomas Wang, Siddharth Karamcheti, Alexander Rush, Douwe Kiela, et al. Obelics: An open web-scale filtered dataset of interleaved image-text documents. *Advances in Neural Information Processing Systems*, 36, 2024. 6

[14] Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. What matters when building vision-language models?, 2024. 2, 4

[15] Beibin Li, Yi Zhang, Sébastien Bubeck, Jeevan Pathuri, and Ishai Menache. Small language models for application interactions: A case study, 2024. 1

[16] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 19730–19742. PMLR, 23–29 Jul 2023. 2

[17] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation, 2022. 2

[18] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of Machine Learning and Systems*, 6:87–100, 2024. 8

[19] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pre-training for visual language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26689–26699, June 2024. 2

[20] Rongmei Lin, Xiang He, Jie Feng, Nasser Zalmout, Yan Liang, Li Xiong, and Xin Luna Dong. Pam: Understanding product images in cross product category attribute extraction. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 3262–3270, New York, NY, USA, 2021. Association for Computing Machinery. 2

[21] Haotian Liu. Llava pretrained dataset, 2024. 6

[22] Haotian Liu. Llava1.5 finetuning data, 2024. 6

[23] Haotian Liu. Llava1.5 finetuning script, 2024. 6

[24] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 26296–26306, June 2024. 2, 4, 6

[25] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916. Curran Associates, Inc., 2023. 2

[26] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. LaVANext, 2024. Accessed on July 14th, 2024. 2, 6

[27] Brandon McKinzie, Zhe Gan, Jean-Philippe Fauconnier, Sam Dodge, Bowen Zhang, Philipp Dufter, Dhruti Shah, Xianzhi Du, Futang Peng, Floris Weers, Anton Belyi, Haotian Zhang, Karanjeet Singh, Doug Kang, Ankur Jain, Hongyu Hè, Max Schwarzer, Tom Gunter, Xiang Kong, Aonan Zhang, Jianyu Wang, Chong Wang, Nan Du, Tao Lei, Sam Wiseman, Guoli Yin, Mark Lee, Zirui Wang, Ruoming Pang, Peter Grasch, Alexander Toshev, and Yinfei Yang. Mm1: Methods, analysis insights from multimodal llm pre-training, 2024. 2, 4

[28] Microsoft. What is Github Copilot? Accessed on July 14th, 2024. 1

[29] Microsoft. Phi-3-vision-128k-instruct, 2024. 5

[30] mistralai. Mistral-7b-instruct-v0.2, 2024. 5

[31] Edwin G. Ng, Bo Pang, Piyush Sharma, and Radu Soricut. Understanding guided image captioning performance across domains. *arXiv preprint arXiv:2012.02339*, 2020. 6

[32] Athanasios N. Nikolakopoulos, Swati Kaul, Siva Karthik Gade, Bella Dubrov, Umit Batur, and Suleiman Ali Khan. Sage: Structured attribute value generation for billion-scale product catalogs, 2023. 2, 7

[33] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world, 2023. 2

[34] 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, 2021. 2

[35] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018. 6

[36] Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8317–8326, 2019. 7

[37] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 6

[38] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. Cambrian-1: A fully open, vision-centric exploration of multimodal llms, 2024. 2

[39] Wenyi Wu, Qi Li, Wenliang Zhong, and Junzhou Huang. Mivc: Multiple instance visual component for visual-language models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 8117–8126, January 2024. 3

[40] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training, 2023. 2, 5

[41] Wanrong Zhu, Jack Hessel, Anas Awadalla, Samir Yitzhak Gadre, Jesse Dodge, Alex Fang, Youngjae Yu, Ludwig Schmidt, William Yang Wang, and Yejin Choi. Multimodal c4: An open, billion-scale corpus of images interleaved with text, 2023. 2

[42] Henry Peng Zou, Vinay Samuel, Yue Zhou, Weizhi Zhang, Liancheng Fang, Zihe Song, Philip S. Yu, and Cornelia Caragea. Implicitave: An open-source dataset and multimodal llms benchmark for implicit attribut value extraction. *arXiv preprint arXiv:2404.15592*, 2024. 1

[43] Henry Peng Zou, Gavin Heqing Yu, Ziwei Fan, Dan Bu, Han Liu, Peng Dai, Dongmei Jia, and Cornelia Caragea. Eiven: Efficient implicit attribute value extraction using multimodal llm. *arXiv preprint arXiv:2404.08886*, 2024. 1, 2