

MTIVE: Multi-Task Image Verification Engine Using Vision-Language Models for E-commerce

Yu-Tong Cao¹ Vishnu Prabhakaran² Arunita Das² Purav Aggarwal² Anoop Saladi²

¹Amazon, Australia ²Amazon, India

{ytcao, visprab, arunita, aggap, saladias}@amazon.com

Abstract

Vision-language models show promise for e-commerce automation but struggle with noisy real-world images and multi-task requirements. We introduce MTIVE, a curriculum learning framework that progressively adapts base models through three stages: continued pre-training on large-scale e-commerce datasets with contrastive learning and diverse dialogue templates, instruction tuning on synthetic data, and modular task-specific expert training. Our architecture uses frozen base weights with stacked LoRA adapters—shared modules for domain knowledge and lightweight task-specific experts—enabling continual learning without catastrophic forgetting. MTIVE outperforms open-source and proprietary baselines in both standard and continual learning settings.

1 Introduction

The rapid growth of e-commerce has created a complex ecosystem where millions of transactions generate vast amounts of visual and textual data across packaging, shipping, delivery, and post-delivery stages. Maintaining customer trust and operational integrity requires verifying image-based evidence collected throughout the product lifecycle—from quality control at fulfillment centers to customer claims for refunds, returns, and concessions. Currently, manual verification processes cannot scale effectively, with only 47% of collected images examined by auditors. This creates significant gaps that bad actors can exploit while also limiting the platform’s ability to efficiently serve legitimate customer needs. As operations scale and visual evidence accumulates, the lack of automated image verification at scale risks both revenue protection and customer-centric policies. Bridging this gap demands robust automated solutions that can analyze visual and textual data together to address critical use cases such as verifying product authenticity,

assessing damages, monitoring quality control, and resolving claims across the entire supply chain.

In the e-commerce domain, Vision-Language Models (VLMs) have been increasingly leveraged for various applications through adaptation of general-purpose models with domain-specific datasets (Fu et al., 2022; Khandelwal et al., 2023; Jia et al., 2023; Trabelsi et al., 2025; Yada et al., 2025). However, there are two major limitations in such works. Firstly, beyond simple domain adaptation on a single task, these platforms require multi-task learning capabilities to simultaneously handle diverse verification scenarios, from damage detection and product matching to attribute extraction and packaging verification, each demanding different types of visual reasoning and accuracy requirements. Secondly, these models remain constrained by their training data distribution, exhibiting limited zero-shot performance on novel tasks. Real-world deployed systems frequently need to expand capabilities across different marketplaces, support new modalities, or upgrade functionalities. The conventional approach of retraining the entire model on expanded datasets introduces several critical challenges: performance regression on previously mastered tasks, task interference, varying data quality, difficulty achieving Pareto-optimal solutions, deployment disruptions, and inability to add, update, or deprecate capabilities without service interruption.

To address these challenges, we present MTIVE (Multi-Task Image Verification Engine), which proposes a curriculum based learning method for adapting general-purpose VLMs to e-commerce multi-task learning, along with a dynamic multi-LoRA (Hu et al., 2021) serving framework for efficient model composition and continual task addition. Our key contributions are:

- We introduce a three-stage curriculum learning approach: (i) task-agnostic pre-training on e-commerce data to learn domain-specific visual

patterns, (ii) instruction fine-tuning for instruction-following capabilities, and (iii) task-specific expert training. Our modular architecture employs frozen base weights with progressively stacked LoRA adapters—shared adapters for common e-commerce knowledge and task-specific experts for specialized capabilities—enabling efficient multi-task learning without catastrophic forgetting.

- We develop a dynamic multi-LoRA serving system on vLLM (Kwon et al., 2023a) supporting efficient batched inference and runtime hot-swapping of expert adapters, enabling seamless addition, update, or removal of verification tasks without service disruption.
- We conduct extensive experiments on real-world e-commerce verification tasks, demonstrating MTIVE’s superior performance against baselines. Through ablation studies, we validate each architectural component and confirm our design motivations for multi-task and continual learning.

2 Related Work

VLMs for E-Commerce. Open-source pre-trained VLMs such as IDEFICS (Laureçon et al., 2024), LLaVA (Liu et al., 2024), Qwen (Bai et al., 2025) and Gemma (Kamath et al., 2025) demonstrate strong multi-modal capabilities but lack domain-specific understanding for e-commerce applications. Prior works address this through domain-specific fine-tuning (Fu et al., 2022; Khandelwal et al., 2023; Jia et al., 2023; Trabelsi et al., 2025; Yada et al., 2025), typically focusing on single tasks with high-quality catalog images for attribute extraction and catalog improvement. VIT-Pro (Prabhakaran et al., 2025) recently extended this to customer feedback images with cost-efficient data generation. However, these approaches employ single-stage training on task-specific datasets without curriculum strategies, limiting their ability to handle the diverse image verification tasks required across the e-commerce supply chain or maintain robust multi-task performance when adapting to new capabilities.

Multi-Task Learning. Multi-Task Learning (MTL) aims to jointly optimize all tasks using a single trained model. In MTL for LLMs, key challenges include task interference, catastrophic forgetting, training instability on mixed datasets, and high retraining costs. LoRA-based MTL methods offer parameter efficiency but encounter instability

on large mixed datasets, as LoRA projects sparse high-dimensional features from different tasks into the same dense low-dimensional space, causing task interference. To address this, existing MoE-based PEFT methods (Wang et al., 2023; Zadouri et al., 2023; Agiza et al., 2024; Yang et al., 2025) train multiple experts jointly on mixed datasets with routing mechanisms to dynamically allocate expert contributions per task. However, single-stage joint training still causes performance degradation from data interference and requires full retraining for new tasks. In contrast, MTIVE adopts a three-stage curriculum paradigm (task-agnostic pre-training, instruction tuning, and task-specific expert training) using stacked LoRA adapters where experts are trained separately rather than jointly for robust multi-task learning. This staged approach mitigates performance degradation from mixed datasets while balancing general and domain-specific capabilities, enabling plug-and-play task addition via hot-swappable experts for seamless production updates without retraining or catastrophic forgetting.

3 MTIVE

In this section, we present a comprehensive framework for adapting general-purpose VLMs to multi-task e-commerce scenarios, comprising two main components: a three-stage curriculum learning method and a dynamic multi-LoRA serving architecture. Our curriculum learning approach, illustrated in Figure 1, consists of three sequential stages. First, we perform task-agnostic pre-training on raw image-text pairs from large-scale e-commerce datasets to achieve e-commerce concept alignment. Second, we continue training the model on synthetically generated instruction-following data to develop instruction-following capabilities. Third, we train multiple task-specific experts on high-quality human-annotated data to improve performance and enable flexible addition of new tasks. Throughout all training stages, the base VLM weights (W^0) remain frozen, and we employ LoRA-based fine-tuning. We denote W^{PT} as the LoRA weights from the pre-training stage, W^{IT} as the LoRA weights from the instruction fine-tuning stage, and $W^{E_1}, W^{E_2}, \dots, W^{E_T}$ as the weight matrices of LoRA experts for T tasks. For production deployment, we introduce a multi-LoRA serving architecture that efficiently handles heterogeneous task batches while supporting hot-swapping of experts without service disruption.

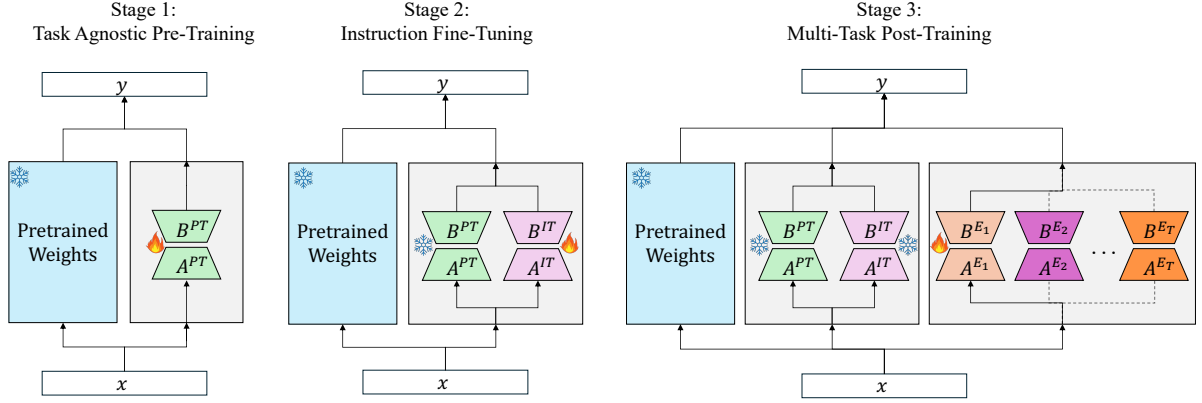


Figure 1: Overview of the three-stage training pipeline for MTIVE. The model is initialized with a frozen pre-trained VLM and progressively adapted through LoRA-based curriculum learning: (1) task-agnostic pre-training for e-commerce concept alignment (W^{PT}), (2) instruction fine-tuning on synthetic data (W^{IT}), and (3) multi-task post-training with task-specific experts (W^{E_1}, \dots, W^{E_T}).

3.1 Task-Agnostic Pre-Training

To balance concept coverage and training efficiency, we sample 2.5M image-text pairs from multiple data sources across the e-commerce order lifecycle: product catalogs, customer complaints, and return/refund conversations, spanning 48 product categories. Product catalog images are high-quality with well-structured text due to listing standards. However, customer-generated sources contain non-standard, in-the-wild data that is inherently noisy—featuring poor lighting, occlusion, blur, and inconsistent framing, with text that is often informal or misaligned with visual content. Despite this noise, leveraging naturally occurring data provides cost-effective domain adaptation that closely matches our target verification tasks, without requiring expensive manual annotations at scale. This stage expands the vocabulary of aligned image-text tokens to the e-commerce domain. During this phase, W^{PT} undergoes training while W^0 remains frozen. The matrix W^{PT} is factorized into low-rank matrices $B^{PT} \in \mathbb{R}^{d \times r}$ and $A^{PT} \in \mathbb{R}^{r \times k}$. For layer i , the output is:

$$y_i = W_i^0 x + \alpha^{PT} B_i^{PT} A_i^{PT} x$$

where α^{PT} scales the adaptation magnitude.

We employ three complementary techniques during pre-training to build robust e-commerce vision-language understanding. First, we construct 6 diverse dialogue templates apart from using raw image-text pairs, enabling the model to learn multi-modal reasoning across varied image orderings and context availability while supporting both generative description and comparative

reasoning between catalog and customer images. Second, we apply noise-robust training through label smoothing (Szegedy et al., 2015) and uniform distribution regularization (Pereyra et al., 2017) to address inherent ambiguities and inconsistencies in customer-generated feedback, preventing overconfident predictions and improving generalization on noisy data. Specifically, our loss function is defined as $\mathcal{L} = (1 - \beta)\mathcal{L}_{CE} + \beta\mathcal{L}_{uniform}$, where \mathcal{L}_{CE} is the standard cross-entropy loss, $\mathcal{L}_{uniform} = -\log(|V|)$ encourages uniform probability distribution over the vocabulary of size $|V|$, and β controls the regularization strength. Third, we incorporate contrastive learning with TF-IDF-based (Sparck Jones, 1972) hard negative mining to teach fine-grained product discrimination, constructing challenging pairs of semantically similar but visually distinct products that enhance the model’s visual discrimination capabilities across professional catalog and customer-submitted images through 4 extra dialogue templates (see Appendix A). Together, these techniques create a comprehensive pre-training regime that balances generative fluency with discriminative precision for e-commerce vision-language understanding.

3.2 Instruction Fine-Tuning

To develop instruction-following capabilities, we convert a portion ($\sim 50K$) of the pre-training dataset containing raw image-text pairs into a visual instruction-following format using an open-source pre-trained VLM via internal APIs. More specifically, we feed the VLM with both the product image and corresponding text from customer feedback and product descriptions (with confidential

information redacted), then prompt it to generate different types of instruction-following data for e-commerce. See Appendix B for the prompt template. During this phase, W^{IT} undergoes training while W^0 and W^{PT} remain frozen. The matrix W^{IT} can be factorized into low-rank matrices $B^{IT} \in \mathbb{R}^{d \times r}$ and $A^{IT} \in \mathbb{R}^{r \times k}$. For layer i , the output y_i corresponding to input x_i computed through this modified weight matrix is expressed as:

$$y_i = W_i^0 x + \alpha^{PT} B_i^{PT} A_i^{PT} x + \alpha^{IT} B_i^{IT} A_i^{IT} x$$

where α^{IT} scales the magnitude of adaptation from the pre-trained model.

3.3 Multi-Task Post-Training

In this stage, the task-specific experts E_1, E_2, \dots, E_T are trained separately on their respective task-specific datasets. LoRA adapters are injected on top of the instruction-tuned model from previous training stage. During this phase, W^{E_t} undergoes training while W^0 , W^{PT} and W^{IT} remain frozen. The matrix W^{E_t} can be factorized into low-rank matrices $B^{E_t} \in \mathbb{R}^{d \times r}$ and $A^{E_t} \in \mathbb{R}^{r \times k}$. For an expert E_t , the output y_i corresponding to input x_i in layer i computed through this modified weight matrix is expressed as:

$$y_i = W_i^0 x + \alpha^{PT} B_i^{PT} A_i^{PT} x + \alpha^{IT} B_i^{IT} A_i^{IT} x + \alpha^{E_t} B_i^{E_t} A_i^{E_t} x$$

where α^{E_t} scales the magnitude of adaptation from the instruction-tuned model.

Since each task-specific expert is trained independently, we can flexibly customize training configurations based on task characteristics: employing different LoRA ranks r_t , selectively applying LoRA to vision encoder, language decoder, projector or all components, injecting adapters into different layer subsets (e.g., attention only vs. attention and FFN), applying different post-training strategies (SFT, DPO, GRPO or MPO), and using task-specific hyperparameters and data sampling strategies. This enables efficient resource allocation, dedicating higher capacity and sophisticated methods to challenging tasks while using simpler configurations for easier ones.

3.4 Dynamic Multi-LoRA Serving Architecture

To support the increasing demand for diverse downstream e-commerce tasks, we build an effi-

cient multi-task serving system that handles multiple LoRA adapters simultaneously. Using the vLLM (Kwon et al., 2023b) framework, the system shares our instruction-tuned base model (from subsection 3.2) across different tasks and activate different LoRA adapters based on incoming requests, allowing multiple tasks to be processed together in the same batch. Each user request is represented as (x_j, t_j) , where x_j is the input and $t_j \in \{1, 2, \dots, T\}$ specifies the target task. For a heterogeneous batch containing n requests $\{(x_1, t_1), (x_2, t_2), \dots, (x_n, t_n)\}$, the output $y_{i,j}$ for request j at layer i is computed as:

$$y_{i,j}(x_j, t_j) = W_i^0 x_j + \alpha^{PT} B_i^{PT} A_i^{PT} x_j + \alpha^{IT} B_i^{IT} A_i^{IT} x_j + \alpha^{E_{t_j}} B_i^{E_{t_j}} A_i^{E_{t_j}} x_j$$

This architecture enables efficient batched inference by sharing the computational cost of the large base model (W^0) and shared adapters (W^{PT} , W^{IT}) across all requests in the batch. Since these weights constitute the majority of model parameters, batching requests from different tasks together achieves significantly higher GPU utilization and throughput compared to processing each task separately. Only the lightweight task-specific expert weights ($W^{E_{t_j}}$) need to be swapped based on each request’s target task. Figure 2 illustrates the multi-LoRA batched inference architecture with dynamic hot-swapping capabilities.

A key advantage of our multi-task serving architecture is the ability to dynamically add new tasks or update existing experts without service interruption. The vLLM framework supports hot-swapping of LoRA adapters, enabling seamless deployment of new tasks in production environments. When a new task needs to be added to the system, we train a new task-specific expert $W^{E_{T+1}}$ on top of the frozen base model (W^0) and shared adapters (W^{PT} , W^{IT}). Once training is complete, the new expert can be loaded into the serving system and immediately made available for inference. Mathematically, for the newly added task $t = T + 1$, the output computation follows the same formulation:

$$y_i(x, T+1) = W_i^0 x + \alpha^{PT} B_i^{PT} A_i^{PT} x + \alpha^{IT} B_i^{IT} A_i^{IT} x + \alpha^{E_{T+1}} B_i^{E_{T+1}} A_i^{E_{T+1}} x$$

Similarly, existing experts can be updated or deprecated without affecting other tasks. To update an expert E_t , we simply replace the corresponding

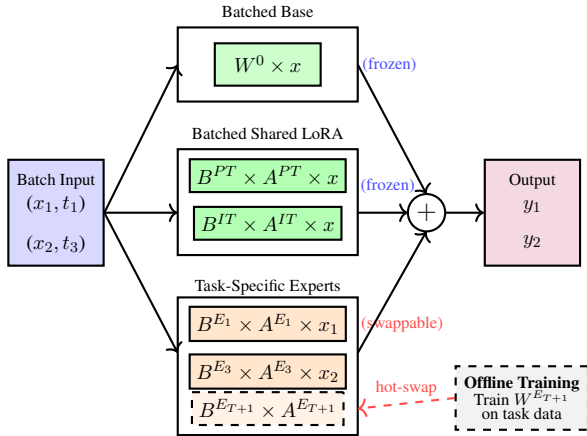


Figure 2: Multi-LoRA Batched Inference with Dynamic Hot-Swapping. The base model and shared LoRA adapters (frozen) process the entire batch in parallel, while task-specific experts can be dynamically hot-swapped. New or updated experts are trained offline and seamlessly deployed without service interruption.

LoRA weights W^{E_t} with the newly trained weights $W^{E'_t}$. To deprecate a task, we remove its associated expert from the serving pool. Throughout these operations, the base model weights (W^0) and shared adapter weights (W^{PT} , W^{IT}) remain unchanged, ensuring that ongoing inference requests for other tasks continue uninterrupted. This capability is essential for e-commerce production systems requiring specialized models for new product categories, continuous model updates, A/B testing of task variants, and flexible seasonal task deployment. The lightweight nature of LoRA adapters (1-2% of base model parameters) enables efficient dynamic deployment with minimal memory overhead.

4 Performance Evaluation

4.1 Experimental Setup

We build upon Qwen2.5-VL-7B (Bai et al., 2025) with public weight initialization. Across all training stages, LoRA is applied to the attention layers of all transformer blocks spanning the vision encoder, language decoder, and connector modules. Images are resized with resolution bounded between $378 \times 28 \times 28$ and $448 \times 28 \times 28$. Pre-training uses LoRA rank $r = 32$ and scaling factor $\alpha = 8$, while instruction tuning and task experts default to $r = 8$ and $\alpha = 8$. We construct EMDIV (Ecommerce Multi-task Dataset for Image Verification), through model-guided human annotation. Our instruction-tuned MTIVE pre-fills candidate labels, which human annotators subsequently verify and

correct, yielding paired outputs that serve both as supervised finetuning targets and as preference data for alignment. EMDIV spans four tasks: Attribute Extraction (AE), Damage Detection (DD), Product Matching (PM), and Expiry Detection (EX), each with up to three variants: *binary* (yes/no classification), *generative* (open-ended reasoning), and *augmented* (LLM-diversified seed samples). In total, EMDIV comprises 350,587 training and 21,434 held-out test samples with no train-test overlap. Appendix C for more details on EMDIV. For evaluation, we employ objective evaluation measuring classification accuracy across task-specific classes, and generative evaluation using Claude-4.5-Sonnet (Anthropic, 2025) as an expert judge to assess response quality against ground truth (see evaluation prompts in Appendix F).

4.2 Multi-Task Learning

Generation-based evaluation We present four MTIVE variants demonstrating our multi-stage training: $MTIVE_{PT}$ (domain pre-training), $MTIVE_{IT}$ (+ instruction tuning), $MTIVE_{SL}$ (+ multi-task with single shared LoRA), and $MTIVE_{MTL}$ (+ multi-task with task-specific LoRA adapters). To validate this design, we compare against three baselines trained on the base model directly: $Base_{SL}$ (single shared LoRA), $Base_{MoE}$ (mixture of experts with 4 experts), and $Base_{MTL}$ (task-specific LoRA adapters). Table 1 reports the model performance of state-of-the-art VLMs, MTL baselines and our MTIVE variants on EMDIV testset across three metrics (accuracy, relevancy, and hallucination). Firstly, Qwen2.5VL-7B demonstrates strong zero-shot performance among open-source VLMs, motivating its selection as our base architecture. Our full model $MTIVE_{MTL}$ achieves the best overall performance across all tasks and metrics. While intermediate stages $MTIVE_{PT}$ and $MTIVE_{IT}$ occasionally underperform the base model, this is expected: continued pre-training optimizes for e-commerce domain understanding using noisy e-commerce raw data corpus, and instruction tuning focuses on learning task-specific formats rather than maximizing immediate performance. However, these foundational stages prove crucial, as $MTIVE_{SL}$ significantly outperforms $Base_{SL}$ and $MTIVE_{MTL}$ similarly surpasses $Base_{MTL}$, demonstrating that domain knowledge and task understanding from earlier stages compound with subsequent multi-task learning. We also evaluate $Base_{MoE}$, where a learned routing mechanism

Table 1: Multi-Task Learning: Quantitative evaluation results on EMDIV (relative to Idefics2-8B).

| Model | AE | | | DD | | | PM | | | EX | | |
|------------------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | Acc.↑ | Rel.↑ | Hal.↓ | Acc.↑ | Rel.↑ | Hal.↓ | Acc.↑ | Rel.↑ | Hal.↓ | Acc.↑ | Rel.↑ | Hal.↓ |
| Idefics3-8B | +10.4 | +5.8 | -2.3 | +10.0 | +11.8 | -8.4 | +8.9 | +23.0 | -17.2 | -7.7 | -3.7 | +12.6 |
| Gemma3-4B | +21.3 | +24.8 | -19.3 | +1.1 | +18.5 | +16.6 | +12.5 | +33.1 | -14.2 | +18.4 | +7.7 | -24.5 |
| Gemma3-12B | +45.9 | +41.5 | -63.8 | +5.3 | +23.8 | +0.1 | +25.7 | +40.4 | -35.4 | +16.0 | +13.2 | -28.2 |
| Qwen2.5VL-3B | +38.2 | +32.4 | -21.7 | +6.0 | +18.8 | -2.3 | +22.6 | +36.3 | -28.3 | -2.0 | +11.1 | -9.3 |
| Qwen2.5VL-7B | +46.1 | +48.2 | -51.7 | +11.7 | +20.2 | -12.7 | +27.9 | +39.9 | -38.1 | +3.6 | +10.7 | -19.8 |
| Qwen2.5VL-7B (ICL) | +51.7 | +52.5 | -58.0 | +13.2 | +22.5 | -13.4 | +31.6 | +38.7 | -38.9 | +4.0 | +11.7 | -20.9 |
| Claude Haiku 4.5 | +33.1 | +28.6 | -15.8 | -1.9 | +25.7 | +16.6 | +25.3 | +32.1 | -36.7 | +12.7 | +16.1 | -34.0 |
| Claude Haiku 4.5 (ICL) | +38.0 | +32.7 | -18.1 | +5.5 | +29.0 | -3.2 | +27.7 | +38.0 | -38.1 | +15.4 | +17.7 | -42.3 |
| Base _{SL} | +55.0 | +58.7 | -45.3 | +17.1 | +21.6 | -7.4 | +33.4 | +37.3 | -38.6 | +30.7 | +18.2 | -45.4 |
| Base _{MoE} | +58.5 | +65.0 | -50.5 | +19.5 | +22.0 | -10.5 | +34.0 | +38.5 | -37.0 | +31.0 | +18.5 | -45.8 |
| Base _{MTL} | +62.8 | +77.5 | -59.3 | +21.0 | +22.6 | -14.0 | +33.9 | +39.2 | -36.4 | +30.7 | +18.2 | -46.1 |
| MTIVE _{PT} | -17.8 | -20.1 | +24.5 | -20.9 | -33.9 | +22.7 | -15.2 | -18.5 | +20.3 | -44.7 | -71.1 | +38.4 |
| MTIVE _{IT} | +5.7 | +7.8 | -9.1 | -1.4 | +19.0 | +13.0 | +3.2 | +8.5 | +10.8 | -10.8 | +12.2 | -12.5 |
| MTIVE _{SL} | +56.2 | +60.4 | -46.9 | +18.3 | +22.1 | -10.4 | +34.2 | +38.5 | -37.4 | +31.3 | +18.4 | -45.8 |
| MTIVE _{MTL} | +64.8 | +78.1 | -61.6 | +23.1 | +23.8 | -14.5 | +34.3 | +40.8 | -39.8 | +31.5 | +19.1 | -47.3 |

dynamically assigns tokens to experts rather than explicitly mapping them to tasks. While *Base_{MoE}* outperforms *Base_{SL}* through sparse expert activation, it underperforms *Base_{MTL}*, highlighting the advantage of explicit task-specific specialization. The consistent progression from single shared LoRA to MoE to task-specific adapters validates our multi-stage multi-LoRA approach, where each component—domain pre-training, instruction tuning, and task-specific adaptation—plays a complementary role in achieving superior e-commerce visual verification performance.

Classification-based evaluation We present the classification-based evaluation results in Table 2. Our best model, *MTIVE_{MTL}*, achieves state-of-the-art performance, outperforming all baselines on both damage detection (+0.48 F1-weighted) and product matching (+0.60 F1-weighted). Notably, *MTIVE_{MTL}* surpasses the strongest baseline (Claude Sonnet 3.7: DD +0.44, PM +0.21) by substantial margins of +0.04 on DD and +0.39 on PM, demonstrating the effectiveness of our multi-task LoRA approach.

Qualitative Results Table 3 illustrates the qualitative performance of models using task-specific instructions to study their generative capability with one product matching example (refer to Table 10 for more examples). General-domain VLMs exhibit limited zero-shot capabilities for domain-specific use cases. Their sub-optimal performance can be attributed to: *a*) limited effectiveness on in-the-wild images with partially visible regions, occlusions & poor-quality, and *b*) limited generalization to out-of-domain and complex visual reasoning tasks. MTIVE bridges this domain gap,

Table 2: Classification Results (F1-weighted score relative to Gemma3-4B)

| Model | DD | PM |
|----------------------------|--------------|--------------|
| <i>Open-source</i> | | |
| Gemma3-4B | * | * |
| Qwen2.5-VL-3B | +0.42 | +0.20 |
| Qwen2.5-VL-7B | +0.44 | +0.21 |
| IDEFICS2-8B | +0.42 | +0.21 |
| <i>Closed-source</i> | | |
| Claude Sonnet 3.7 | +0.42 | +0.37 |
| Claude Haiku 4.5 | +0.39 | +0.20 |
| <i>Our Model</i> | | |
| MTIVE_{MTL} | +0.48 | +0.60 |


showing promising visual recognition and reasoning capabilities for the e-commerce domain.

4.3 Continual Task Learning

To assess flexibility under sequential task expansion, we extend MTIVE beyond the four existing tasks (AE, DD, PM, EX) to incorporate two novel tasks: Packaging Type (PT), a 6-class classification task with 2,859 training and 340 test samples, and Product Placement (PP), a 4-class classification task with 1,418 training and 203 test samples. We compare against: *Base_{SL}*, the single shared LoRA before expansion; *Base_{SL}+Replay*, the same adapter extended to new tasks with varying replay ratios (0%, 1%, 100%) of previous task data; and *Base_{MTL}*, independent task-specific LoRA adapters requiring no retraining on old tasks. Evaluation uses classification accuracy for binary tasks (AE, DD, PM), exact match for EX, and multi-class accuracy for PT and PP. Table 4 presents relative gains over the zero-shot baseline. *Base_{SL}* performs well on existing tasks (DD: +10.8, EX: +16.8) but

Table 3: A challenging example from EMDIV dataset to compare visual reasoning and information extraction capabilities.

Product Matching



User: Product Description: Greenbrew Disinfectant Wet Wipes (Lemon) - 10 Single Sachets (Pack Of 2). Does the product description match with the image?

Groundtruth: The product in the image is Greenbrew Disinfectant Wet Wipes (Lemon) - 10 Single Sachets. The product description is for the same product but in a pack of 2. The pack size cannot be determined from the image.

ClaudeV4: The product description **accurately reflects what is shown in the image and matches the text from the packaging.** All key details are consistent.

Qwen2.5VL: Yes, the product description matches with the image. The image shows a box containing disinfecting wet wipes with lemon fragrance **in a pack of 10 single sachets.**

MTIVEMTL: The image shows Greenbrew Disinfectant Wet Wipes (Lemon) - 10 Single Sachets while the product description is for Pack Of 2. The quantity cannot be identified from the image.

Table 4: Continual Task Learning: Classification accuracy (relative to zero-shot Claude-4.5 Sonnet).

| Config. | Existing | | | | New | |
|--------------------------------|-------------|--------------|-------------|--------------|-------------|--------------|
| | AE | DD | PM | EX | PT | PP |
| Base _{SL} | +7.2 | +10.8 | +7.1 | +16.8 | -10.1 | -14.5 |
| Base _{SL} +Rep.(0%) | -4.3 | -10.2 | -7.5 | -9.2 | +2.3 | +3.7 |
| Base _{SL} +Rep.(1%) | +3.3 | +2.1 | +5.4 | +8.7 | +2.2 | +3.5 |
| Base _{SL} +Rep.(100%) | +6.4 | +10.2 | +6.5 | +16.3 | +2.5 | +4.4 |
| Base _{MTL} | +5.1 | +4.2 | +3.7 | +10.5 | +3.5 | +5.4 |
| MTIVEMTL | +9.2 | +12.4 | +8.3 | +18.1 | +5.2 | +11.4 |

degrades on new tasks (PT: -10.1, PP: -14.5). Extending it without replay causes catastrophic forgetting (DD: -10.2, EX: -9.2), while increasing replay from 1% to 100% progressively recovers existing task performance but yields only marginal new task gains (PP: +4.4 at best) with full retraining cost. *Base_{MTL}* avoids forgetting but lacks shared representations, limiting gains across both existing and new tasks. *MTIVEMTL* achieves the highest gains on all tasks (AE: +9.2, DD: +12.4, EX: +18.1, PP: +11.4), demonstrating that our multi-stage design preserves prior knowledge while enabling strong adaptation to novel tasks without replay buffers. See detailed operational analysis in [Appendix D](#).

5 Industry Impact of MTIVE

Manual investigations of multimodal customer queries, such as verifying transit damage, product mismatches, and quantity assurance remain labor-intensive and unsustainable at modern commerce velocity. MTIVE addresses this by supporting incremental task expansion without disrupting existing capabilities, enabling the gradual automation of diverse investigations while preserving prior task performance. A four-week shadow-mode de-

ployment in a co-pilot setting across four operational tasks (damage detection, product matching, attribute extraction, and expired product detection) demonstrated significant improvement in investigation efficiency and greater decision consistency. These findings show that MTIVE streamlines multimodal investigations, leading to substantial operational savings. The framework serves as a scalable layer that enhances human decision-making while ensuring consistent performance during incremental task expansion.

6 Conclusion

We present MTIVE, a comprehensive framework for adapting VLMs to multi-task e-commerce scenarios through three-stage curriculum learning: task-agnostic pre-training, instruction fine-tuning with synthetic data, and multi-task post-training with task-specific LoRA experts. Our dynamic multi-LoRA serving architecture enables efficient batched inference with hot-swappable experts, achieving state-of-the-art performance. Results validate each component and demonstrate continual learning with minimal catastrophic forgetting, enabling adaptation to new business needs. Experimental results validate the framework’s continual learning capability, successfully acquiring new tasks with minimal catastrophic forgetting, making it particularly suitable for production environments requiring continuous adaptation to evolving business needs.

References

- Ahmed Agiza, Marina Neseem, and Sherief Reda. 2024. [Mtlora: A low-rank adaptation approach for efficient multi-task learning](#). *Preprint*, arXiv:2403.20320.
- AI Anthropic. 2025. System card: Claude opus 4 & claude sonnet 4. *Claude-4 Model Card*.
- Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibao Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.
- Jinmiao Fu, Shaoyuan Xu, Huidong Liu, Yang Liu, Ning Xie, Chien-Chih Wang, Jia Liu, Yi Sun, and Bryan Wang. 2022. [Cma-clip: Cross-modality attention clip for text-image classification](#). In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 2846–2850.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. [Lora: Low-rank adaptation of large language models](#). *Preprint*, arXiv:2106.09685.
- Qinjin Jia, Yang Liu, Daoping Wu, Shaoyuan Xu, Huidong Liu, Jinmiao Fu, Roland Vollgraf, and Bryan Wang. 2023. [KG-FLIP: Knowledge-guided fashion-domain language-image pre-training for E-commerce](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 81–88, Toronto, Canada. Association for Computational Linguistics.
- Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.
- Anant Khandelwal, Happy Mittal, Shreyas Kulkarni, and Deepak Gupta. 2023. [Large scale generative multimodal attribute extraction for E-commerce attributes](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 305–312, Toronto, Canada. Association for Computational Linguistics.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023a. [Efficient memory management for large language model serving with pagedattention](#). *Preprint*, arXiv:2309.06180.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023b. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.
- Hugo Laurençon, Andrés Marafioti, Victor Sanh, and Léo Tronchon. 2024. Building and better understanding vision-language models: insights and future directions. *arXiv preprint arXiv:2408.12637*.
- Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024. [Llava-next: Improved reasoning, ocr, and world knowledge](#).
- Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*.
- Vishnu Prabhakaran, Purav Aggarwal, Vishrui Kulkshreshtha, Arunita Das, Sahini Venkata Sitaram Sruti, and Anoop Saladi. 2025. Vit-pro: Visual instruction tuning for product images. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 3: Industry Track)*, pages 695–707.
- Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.
- C Szegedy, V Vanhoucke, S Ioffe, J Shlens, and Z Wojna. 2015. Rethinking the inception architecture for computer vision. arxiv preprint arxiv: 1512.00567. *Descargado de https://arxiv.org/abs/1512.00567 doi*, 10.
- Ameni Trabelsi, Maria Zontak, Yiming Qian, Brian Jackson, Suleiman Khan, and Umit Batur. 2025. [What Matters When Building Vision Language Models for Product Image Analysis?](#). In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, pages 1282–1291, Los Alamitos, CA, USA. IEEE Computer Society.
- Yiming Wang, Yu Lin, Xiaodong Zeng, and Guannan Zhang. 2023. [Multilora: Democratizing lora for better multi-task learning](#). *Preprint*, arXiv:2311.11501.
- Yuki Yada, Sho Akiyama, Ryo Watanabe, Yuta Ueno, Yusuke Shido, and Andre Rusli. 2025. Improving visual recommendation on e-commerce platforms using vision-language models. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, pages 975–978.
- Yaming Yang, Dilxat Muhtar, Yelong Shen, Yuefeng Zhan, Jianfeng Liu, Yujing Wang, Hao Sun, Denny Deng, Feng Sun, Qi Zhang, Weizhu Chen, and Yunhai Tong. 2025. [Mtl-lora: Low-rank adaptation for multi-task learning](#). *Preprint*, arXiv:2410.09437.
- Ted Zadouri, Ahmet Üstün, Arash Ahmadian, Beyza Ermiş, Acyr Locatelli, and Sara Hooker. 2023. [Pushing mixture of experts to the limit: Extremely parameter efficient moe for instruction tuning](#). *Preprint*, arXiv:2309.05444.

A Pre-training Dialogue Templates

Table 5 summarizes all ten dialogue templates used during pre-training, sampled with equal probability.

| Type | Input → Output |
|-------------|---|
| Generative | Catalog + Desc. + Feedback imgs → Feedback text |
| Generative | Catalog imgs → Desc., Feedback imgs → Text |
| Generative | Catalog imgs → Product description |
| Generative | Catalog + Feedback imgs → Feedback text |
| Generative | Feedback imgs → Feedback text |
| Generative | Shuffled imgs → Product description |
| Contrastive | Catalog + Hard neg. catalog → Yes/No |
| Contrastive | Catalog + Hard neg. feedback → Yes/No |
| Contrastive | Catalog + Matching feedback → Yes/No |
| Contrastive | Hard neg. feedback + catalog → Yes/No |

Table 5: Pre-training dialogue templates. First six are generative; last four are contrastive with TF-IDF hard negatives.

B Instruction Data Generation Prompt

Table 6 shows the prompt template used for generating visual instruction-following data.

C EMDIV dataset

Table 7 summarizes the EMDIV statistics across four tasks: attribute extraction (AE) from customer images, damage detection (DD) for product damage identification, product matching (PM) for image-product verification, and expiry detection (EX) for manufacturing and expiry date recognition.

D Operational Analysis

We analyze MTIVE’s operational characteristics across three dimensions: system-level scalability, inference latency, and deployment considerations.

System Comparison. Table 8 compares MTIVE against baseline configurations across key deployment properties. $MTIVE_{MTL}$ uniquely combines incremental task integration with architectural modularity while maintaining low task interference, without requiring data re-balancing or replay buffers. Monolithic configurations ($Base_{SL}$, $Base_{SL}+Replay$) necessitate retraining shared parameters for every new task, incurring high interference risk and data management overhead. $Base_{MTL}$ offers modularity but lacks shared representations, limiting cross-task transfer.

Scalability and Training Cost. $MTIVE_{MTL}$ supports incremental task addition while keeping previously deployed weights frozen, eliminating catastrophic forgetting risk and replay overhead. While

Prompt template for visual instruction generation

```
TASK: Generate visual QA pairs for e-commerce
product images.

INPUT:
- Image: Product photo (catalog/customer-submitted)
- Text: Customer feedback or product metadata

REQUIREMENTS:
• Create visually-grounded question-answer pairs
• Cover: product type, quantity, attributes,
condition, packaging, spatial relations
• Acknowledge uncertainty when information is
insufficient
• Use appropriate formats: lists, JSON,
short/detailed text
(...additional requirements omitted)

Customer Context:
<context_1>{CONTEXT_1}</context_1>

Product Metadata:
<context_2>{CONTEXT_2}</context_2>

Examples:
<examples>
...
...
</examples>

Assistant:
```

Table 6: Prompt template for generating visual instruction-following data

| Task | Variant | Train | Test |
|--------------|---------|----------------|---------------|
| AE | Gen. | 3,782 | 1,000 |
| AE | Aug. | 7,168 | 1,911 |
| DD | Gen. | 30,325 | 1,000 |
| DD | Aug. | 118,191 | 3,905 |
| DD | Bin. | 30,325 | 1,000 |
| PM | Gen. | 9,000 | 1,000 |
| PM | Aug. | 34,257 | 3,831 |
| PM | Bin. | 9,000 | 1,000 |
| EX | Gen. | 23,581 | 1,292 |
| EX | Aug. | 45,889 | 2,525 |
| Total | | 350,587 | 21,434 |

Table 7: EMDIV dataset statistics across tasks and variants.

retraining $Base_{SL}$ on all tasks (350k samples) requires ≈ 120 hours on a G5.48xlarge instance, adding a new task to $MTIVE_{MTL}$ scales only with the novel data volume, typically requiring 2–24 hours. Architectural decoupling further enables live module addition without full service redeployment.

Inference Latency. Table 9 reports average inference latency per request across configurations using 100k samples across five tasks. Adapter-based methods were evaluated on a local A10G (G5.48xlarge) instance using vLLM (batch size=1); the zero-shot baseline reflects end-to-end AWS Bedrock response time including network over-

Table 8: System-level comparison across deployment dimensions. ✓ = supported; ✗ = not supported.

| Property | Zero-shot | Base _{MTL} | Base _{SL} | +Replay | MTIVE _{MTL} |
|-----------------------------------|-----------|---------------------|--------------------|---------|----------------------|
| 1) Incremental Task Integration | ✓ | ✓ | ✗ | ✗ | ✓ |
| 2) Susceptibility to Interference | Low | Low | High | High | Low |
| 3) Requires Data Re-balancing | ✗ | ✗ | ✓ | ✓ | ✗ |
| 4) Relative Inference Latency | High | Low | Very Low | Low | Low |
| 5) Architectural Modularity | ✗ | ✓ | ✗ | ✗ | ✓ |

Table 9: Inference latency (ms) across configurations.

| Config. | Env. | P50 | P99 |
|----------------------------|---------|-----|-----|
| Claude4.5-Sonnet | Bedrock | 245 | 410 |
| Base _{MTL} | A10G | 112 | 138 |
| Base _{SL} | A10G | 98 | 122 |
| Base _{SL} +Replay | A10G | 101 | 125 |
| MTIVE _{MTL} | A10G | 115 | 141 |

head. MTIVE_{MTL} maintains latency comparable to Base_{MTL} (P50: 115ms vs. 112ms), with the shared hierarchy incurring only modest overhead. Base_{SL} achieves lower raw latency (P50: 98ms) but at the cost of modularity and retraining requirements.

Memory Footprint and Scaling Limits. Deploying an 8B-parameter MTIVE model (FP16) on an A10G GPU requires ≈ 17 GB of the 24GB available, with KV cache reserving 1–2GB at 1 TPS, leaving ≈ 5 GB headroom. The shared adapter (DoRA, $r=16$, targeting attention blocks) occupies ≈ 27 MB, while each task-specific adapter ($r=4$, targeting MLP and output layers) requires ≈ 18 MB. An initial 4-task deployment (≈ 99 MB) scales to ≈ 135 MB with two additional tasks. Although this remains small relative to available headroom, accounting for allocator fragmentation, CUDA graph capture, and asynchronous loading buffers, we restrict the active adapter pool to a 1.5–2GB managed buffer, supporting ≈ 50 task-specific adapters per GPU before tail-latency instability becomes a concern. Beyond this threshold, less frequent adapters can be offloaded to CPU memory and swapped on-demand with a minor first-access latency penalty.

E Qualitative Results

Table 10 illustrates the qualitative performance of models using task-specific instructions to study their generative capability. General-domain VLMs exhibit limited zero-shot capabilities for domain-specific use cases. Their sub-optimal performance can be attributed to: *a*) limited effectiveness on in-the-wild images with partially visible regions, occlusions & poor-quality, and *b*) limited generalization to out-of-domain and complex visual rea-

soning tasks. MTIVE bridges this domain gap, showing promising visual recognition and reasoning capabilities for the e-commerce domain.

F LLM Judge Prompt Templates

To evaluate the models on the generative-benchmark, we use Claude Sonnet 4.5 as the LLM judge. The prompts we provide to the judge to evaluate the relevancy and accuracy is shown in Table 12. Similarly, the prompts to determine hallucination is shown in Table 11.

G Ethics Statement

This research utilizes e-commerce datasets derived from customer refund and return claims, as well as internal product catalogs, all obtained with requisite consent. To ensure privacy and prevent misuse, all data underwent a rigorous de-identification process to redact personally identifiable information (PII). Wherever human annotations were used, annotators were recruited and compensated in strict accordance with industry wage standards. The operational impact of the MTIVE framework was evaluated via a "shadow-mode" deployment. During this phase, the model operated in a "co-pilot" capacity, where all decisions were vetted by domain experts to mitigate risk and ensure high-fidelity outcomes. This study was conducted with full objectivity and adheres to the ACL Code of Ethics and the professional conduct guidelines for AI research

Table 10: Challenging examples from EMDIV dataset to compare visual reasoning and information extraction capabilities.

Product Matching



User: Product Description: Greenbrrew Disinfectant Wet Wipes (Lemon) - 10 Single Sachets (Pack Of 2). Does the product description match with the image?

Groundtruth: The product in the image is Greenbrrew Disinfectant Wet Wipes (Lemon) - 10 Single Sachets. The product description is for the same product but in a pack of 2. The pack size cannot be determined from the image.

ClaudeV4: The product description **accurately reflects what is shown in the image and matches the text from the packaging.** All key details are consistent.

Qwen2.5VL: Yes, the product description matches with the image. The image shows a box containing disinfecting wet wipes with lemon fragrance **in a pack of 10 single sachets.**

MTIVE_{MTL}: The image shows Greenbrrew Disinfectant Wet Wipes (Lemon) - 10 Single Sachets while the product description is for Pack Of 2. The quantity cannot be identified from the image.

Damage Detection



User: What's the product present in the picture and is it damaged?

Groundtruth: The image shows a face steamer. It is in good condition.

ClaudeV4: This appears to be a pink **silicone bowl or container, likely a collapsible/foldable storage bowl.** **This product should not be used in its current condition, as the damage could affect its functionality.**

Qwen2.5VL: The product appears to be a **menstrual cup.** The image shows a pink **menstrual cup placed inside a larger pink container.** There doesn't seem to be any visible damage.

MTIVE_{MTL}: The image shows a pink colored facial steamer. The product is in good condition.

Damage Detection



User: Please use the image to extract product information in the following JSON format (note xxx is placeholder, if the provided information is insufficient, put 'Unanswerable' instead). Format: {'ExpiryDate': 'xxx', 'ManufacturingDate': 'xxx', 'TimeValid': 'xxx'}

Groundtruth: {"ExpiryDate": "11NOV.22", "ManufacturingDate": "14AUG.22", "TimeValid": "Unanswerable"}"

ClaudeV4: {"ExpiryDate": "Unanswerable", "ManufacturingDate": "14AUG22", "TimeValid": "Unanswerable"}

Qwen2.5VL: {"ExpiryDate": "11NOV:22", "ManufacturingDate": "14AUG:22", "TimeValid": "25B.M2 04:33"}

MTIVE_{MTL}: {"ExpiryDate": "11NOV.22", "ManufacturingDate": "14AUG.22", "TimeValid": "Unanswerable"}"

Prompt template to measure hallucination

User: You are an expert annotator tasked with identifying hallucinations in the responses of a model following the guidelines. You will analyze the responses segment by segment and classify them based on their factual accuracy and visual verifiability.

INPUT:
Image: {image}
Question regarding the images: {question}
Ground truth for the question: {answer}
The prediction from a model for the question: {model_output}

TASK:
Analyze the given response by breaking it into semantic segments and classify each segment into one of three categories:

1. ACCURATE: Factual statements directly verifiable from the image
2. ANALYSIS: Reasonable interpretations or subjective observations
3. INACCURATE: Claims contradicting the image or about non-visible elements

ANNOTATION GUIDELINES:

- ACCURATE segments must:
 - * Describe visually present elements
 - * State correct attributes (color, size, number, position)
 - * Make verifiable factual claims
 - * Describe visible spatial relationships
 - * State obvious visible actions/states
- ANALYSIS segments include:
 - * Reasonable interpretations
 - * Style/mood/atmosphere observations
 - * Plausible contextual assumptions
 - * Subjective assessments
 - * General categorizations
- INACCURATE segments contain:
 - * Claims about non-visible elements
 - * Wrong attributes or relationships
 - * Imagined actions/states
 - * Unverifiable temporal claims
 - * Speculations about hidden aspects

PROCESS:

1. Split the description into coherent semantic segments
2. For each segment:
 - Identify the specific claims made
 - Verify against the image
 - Apply appropriate label
 - Provide brief justification
 - Give a confidence score in the range of [1-10], reflecting certainty in annotation based on visual evidence
3. Calculate segment statistics
4. Format output as specified JSON

REQUIRED OUTPUT FORMAT:
(output format omitted...)

Assistant:

Table 11: Prompt template to measure hallucination in model outputs with an LLM-Judge.

Prompt template to measure accuracy and relevancy

User: Given a customer image, question, ground-truth answer, and model prediction, your task is to evaluate the quality of the model's predicted answer.

INPUT:
Image: {image}
Question regarding the images: {question}
Ground truth for the question: {answer}
The prediction from a model for the question: {model_output}

TASK:
As a smart teacher, after examining the image and the ground truth, please score the given prediction on a scale of [1, 10] based on these criteria:

1. RELEVANCY: Assess whether the response directly follows the instruction. If the model doesn't address all parts of the question, reduce its relevancy score. Enclose your relevancy score in <rel></rel> tags.
2. ACCURACY: Evaluate whether the response is accurate with respect to the image content. Enclose your accuracy score in <acc></acc> tags.

ANNOTATION GUIDELINES:

1. Provide reasoning for your ratings. Enclose your justification in <jus></jus> tags.
2. When evaluating the prediction, prioritize what is actually visible in the image. Note that the ground truth may not capture all attributes present in the image.
3. RESPOND ONLY WITH JSON, do not add anything else in prefix or suffix to your response.

REQUIRED OUTPUT FORMAT:
(output format omitted...)

Assistant:

Table 12: Prompt template to measure accuracy and relevancy in model outputs with an LLM-Judge.