

PUMGPT: A Large Vision-Language Model for Product Understanding

Anonymous ACL submission

Abstract

E-commerce platforms benefit from accurate product understanding to enhance sellers' experience and operational efficiency. Traditional methods often focus on isolated tasks such as attribute extraction or categorization, posing adaptability issues to evolving tasks and leading to usability challenges with noisy data from the internet. Current Large Vision Language Models (LVLMs) lack domain-specific fine-tuning, thus falling short in precision and instruction following. To address these issues, we introduce **PUMGPT**, the first e-commerce specialized LVLM designed for multi-modal product understanding tasks. We collected and curated a dataset of over one million products from AliExpress, filtering out non-inferable attributes using a universal hallucination detection framework, resulting in 663k high-quality data samples. **PUMGPT** focuses on five essential tasks aimed at enhancing workflows for e-commerce platforms and retailers. We also introduce **PUMBENCH**, a benchmark to evaluate product understanding across LVLMs. Our experiments show that **PUMGPT** outperforms five open-source LVLMs and GPT-4V and a non-LVLM baseline in product understanding tasks. We also conduct extensive analytical experiments to delve deeply into the superiority of PUMGPT, demonstrating the necessity for a specialized model in the e-commerce domain.¹

1 Introduction

E-commerce platforms extensively rely on a deep understanding of products to boost online shopping experiences. As is shown in Figure 1, for instance, given a product image, the ability to automatically generate appealing caption, accurately categorize the product and extract its attributes not only improves product recommendation(Le and Lauw, 2021; Sun et al., 2020) and product search(Ahuja

¹We will release the code, model weight, and test set and will release the training set as long as it passes a content review. For now one-tenth of the dataset are available.

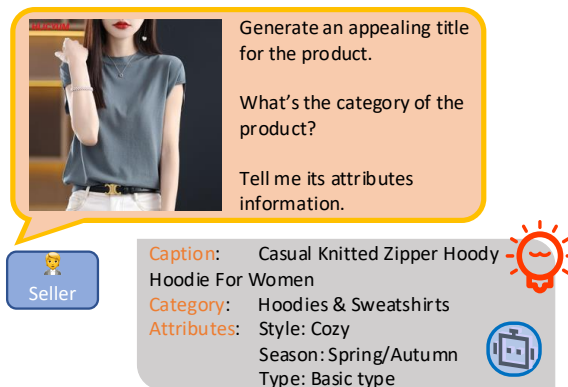


Figure 1: A glimpse on PUMGPT in product understanding.

et al., 2020; Ai et al., 2017) on platforms but also facilitates retailers to launch and update their goods with substantial time savings.

Nevertheless, traditional methods typically focus only on a subset of tasks within a series of product understanding tasks. For instance, they may solely address product attribute extraction(Shinzato et al., 2022; Yan et al., 2021; Zou et al., 2024) or categorization tasks(Lin et al., 2021). Training a specific model for each task proves challenging to adapt to ever-evolving tasks and new products and diminishes usability. Moreover, the product attribute data scraped from the Internet contains a significant amount of noise(Wang et al., 2020; Zhu et al., 2020; Yang et al., 2022). For example, certain attribute values cannot be inferred from the product captions and images since some retailers might supplement the attributes with information not present in the images or captions. Directly training models with such dirty samples can lead to severe hallucination problems(Zhu et al., 2024) in the models. Finally, the suite of product understanding tasks constitutes a multi-modal problem. While current research on Large Vision Language Models (LVLMs)(Bai et al., 2023; Dai et al., 2024; Zhu et al., 2023a; Liu et al., 2023c; Ye et al., 2023) can accomplish

these tasks to some extent, their lack of domain knowledge in e-commerce platforms and still weak instruction following capabilities make them fall short of meeting practical requirements.

To tackle these issues, we present **PUMGPT**, a large vision-language model expert for a series of multi-modal product understanding tasks. To be specific, we collect more than one million product data from the AliExpress platform², including product images, captions, categories, and lists of attributes. To filter out those attributes that cannot be inferred from product images and captions, we propose a universal hallucination detection framework utilizing multi-expert collaboration. Through the thorough hallucinated attributes filtering, we obtain about 663k data for training. Subsequently, we carefully curate five tasks that can help speed up both e-commerce platforms’ and retailers’ workflow. We also introduce **PUMBENCH**, a benchmark covering these product understanding tasks to best evaluate the existing large vision-language models and our **PUMGPT** in the aspect of product understanding. Extensive experiments show the **PUMGPT** outperforms the non-LVLM baseline, 5 open-sourced LVLMs, and GPT-4V(Achiam et al., 2023), the most powerful LVLM for now. And it proves the necessity of a specialized large vision language model for e-commerce.

Our contributions can be summarized as follows:

- We introduce **PUMGPT**, the first e-commerce LVLM for a series of product understanding tasks along with an 663k high-quality product dataset with hallucination filtered.
- We present a universal hallucination detection framework utilizing multi-expert collaboration to detect and filter the inconsistent attributes in the dataset without any labor force.
- Extensive experiments demonstrate the remarkable performance of our **PUMGPT** in **PUMBENCH** over several LVLMs, including GPT-4V.

2 Related Works

Vision-Language Models. Recent advancements have shown significant success in leveraging large language models for vision-language tasks. Equipped with a strong visual encoder, large vision language models(Alayrac et al., 2022; Li et al.,

2023a; Huang et al., 2023b; Driess et al., 2023) achieve alignment between vision and text representations, creating a comprehensive interface for multi-modal input. Commercial models like GPT-4(Achiam et al., 2023) have demonstrated outstanding visual reasoning abilities across diverse vision-linguistic tasks. Increasing model sizes raise computational complexity and training data demands, prompting recent studies to explore efficient fine-tuning methodologies for large vision-language models(Zhu et al., 2023a; Ye et al., 2023; Zhang et al., 2023a). Moreover, the pipeline for pretraining and instruction tuning has emerged as a new paradigm for LVLMs(Liu et al., 2023c; Bai et al., 2023; Dai et al., 2024). However, these models often lack strict adherence to instructions, hampering their usability in large-scale e-commerce scenarios. Our **PUMGPT** is an expert LVLM specifically trained for product understanding tasks, ideally suited for the e-commerce context.

Product Understanding Tasks. Product understanding tasks encompass a variety of sub-tasks. Some studies focus on attribute extraction only with text information.(Zheng et al., 2018; Xu et al., 2019; Yan et al., 2021; Shinzato et al., 2022). Recent research has incorporated visual information from product images to enhance attribute extraction performance (Lin et al., 2021; Zhu et al., 2020; Zhang et al., 2023b; Yang et al., 2022; Zou et al., 2024). The additional visual data enriches the model’s comprehension and extraction capabilities. Besides, other product understanding tasks such as product captioning (Atıcı and İlhan Omurca, 2021), product classification (Bonnett, 2016; Liu et al., 2023a), and even low-level tasks such as retrieval and clustering (Zhan et al., 2021; Dong et al., 2021) have also been explored. However, these solutions typically necessitate training separate models for each task. In contrast, as we compare some product datasets in Table 1, we integrate various product understanding tasks and ensure both quality and scale of the training set with an automated ‘DeHallu’ process to build our **PUMGPT**.

Hallucination Detection. LVLM integrates the capabilities of LLMs and demonstrates strong performance on vision-language tasks; however, it is also affected by LLMs, resulting in hallucination(Tu et al., 2023; Huang et al., 2023a; Zhu et al., 2023b). Therefore, considerable work has focused on researching hallucination detection and mitigation for LVLMs. However, some studies rely on commercial models such as GPT-4V (Xiao et al.,

²<https://www.aliexpress.com/>

Dataset	Language	Quantity	Task	Category	Attribute(N/V)	DeHallu
Product-1M(Zhan et al., 2021)	Chinese	1M	RET	458	N/A	N/A
MEP-3M(Liu et al., 2023a)	Chinese	3M	CLS	599	N/A	N/A
M5Product(Dong et al., 2021)	Chinese	6M	RET/CLS/CLu	6,232	5.6k/24M	N/A
ImplicitAVE(Zou et al., 2024)	English	68.6k	AVE	5 (domains)	25/158	Human
PumGPT(ours)	English	663k	CG/CC/AI/AC/CMC	4,598	11k/48k	Automation

Table 1: The comparison to the previous large-scale e-commerce datasets, where RET for retrieval, CLS for classification, CLu for clustering, and AVE for attribute value extraction. CG/CC/AI/AC/CMC are caption generation/caption completion/attribute inference/attribute correction/category multi-choice. N/V for names/values. DeHallu means hallucination filtering process.

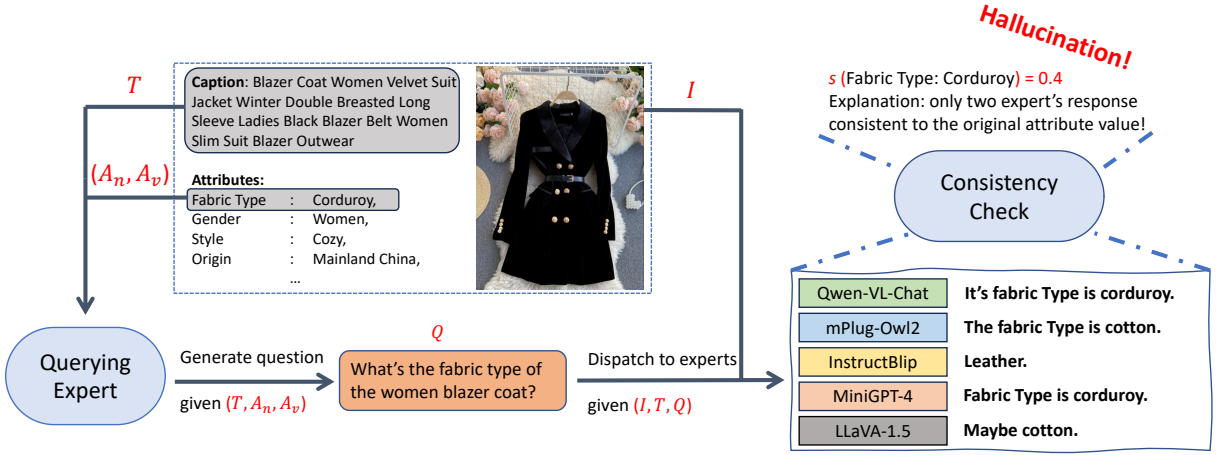


Figure 2: The overview of our proposed hallucination detection framework. We first generate attribute questions according to the product information and dispatch them to different experts to answer. Finally, we employ a judge model to check whether the majority of the answers are consistent with the reference. All the models require no training.

2024; Liu et al., 2023b; Zhao et al., 2023; Zhou et al., 2024) or focus on object-level hallucination detection (Li et al., 2023b; Gunjal et al., 2023). In contrast, we aim to utilize multiple open-source experts for collaborative detection and filtering of product attribute data.

3 PUMGPT

3.1 Data Collection

For sellers, an ideal process for listing products only needs to upload the product images. The system would then automatically generate attractive product titles and compile a series of product attributes for customer reference. The seller would only need to perform a final review and add any additional details if necessary. To achieve this, we gathered a total of about 1 million product entries officially authorized from the AliExpress platform. We sampled top-selling items from various leaf categories (with more extensive sampling from

categories with a higher number of products) to create a relatively high-quality collection. Each product entry contains an image, a caption, the product category, and product attributes. Each attribute consists of an attribute name and a corresponding attribute value. The leaf category is the finest in the taxonomy. For example, under (Automobiles & Motorcycles > Automobiles, Parts & Accessories > Auto Parts > Automobiles Filters > Frequency-separating filters) we ultimately selected the Frequency-separating filters in our dataset. Table 2 demonstrates the statistical results of the initial collected raw data.

3.2 Hallucination Filtering

The initial dataset acquired from the Internet contains substantial noise stemming from multiple factors: many items lack essential product information, such as missing key attributes, making them unsuitable for training. Additionally, certain attributes might either complement product descrip-

Statistical Item	Raw #	Clean #
Products	996,350	663,330
Attributes	10,729,585	1,484,948
Attribute names	12,013	11,291
Attribute values	59,669	48,448
Categories	7,084	4,598

Table 2: The statistical results of the raw collected data and cleaned data. We report the unique items.

tions and images or conflict with other information sources due to sellers’ subjectivity. Consequently, models trained on such datasets might generate inaccuracies during inference. To mitigate this, we propose a universal hallucination detection framework aimed at filtering out noisy samples from a dataset containing approximately one million entries. This framework leverages multi-expert collaboration to identify inconsistent attributes without manual intervention. Contemporary Large Vision Language Models (LVLMs) are pre-trained and fine-tuned on diverse datasets with varying architectures, resulting in significant variability in inference behaviors. Despite these differences, LVLMs tend to align on tasks requiring common knowledge or reasoning, while diverging on ambiguous queries. This property can be leveraged to detect inconsistencies in product datasets, especially where attributes misalign with descriptions and images. By using distinct LVLMs with different knowledge bases, more consistent responses can be obtained for accurate attribute values, while varied responses signal mismatched or supplementary information or subjective attributes.

As is shown in Figure 2, we selected five LVLMs as experts in hallucination detection: $\mathcal{E} = \{\text{Qwen-VL-Chat}(\text{Bai et al., 2023}), \text{MiniGPT-4}(\text{Zhu et al., 2023a}), \text{InstructBLIP}(\text{Dai et al., 2024}), \text{mPLUG-Owl2}(\text{Ye et al., 2023}), \text{LLaVA}(\text{Liu et al., 2023c})\}$. After removing samples with missing information, a standard sample $S = (I, T, C, A_n, A_v)$ is obtained, where I represents the product image, T the product title, C the product category, A_n the attribute name, and A_v the attribute value. For each attribute pair (A_n, A_v) to be queried, a specific attribute question is needed. Since template-based question generation cannot provide an exact question on the value (e.g. Given the Bluetooth attribute, its value might be a version number or yes/no indicating whether the product supports), we employ an LLM to serve as a

querying expert. Due to considerations of time and performance, we choose the Vicuna-13B(Chiang et al., 2023) to generate attribute questions $Q = \text{Vicuna}(P_q, T, A_n, A_v)$. The prompt P_q for generating questions is shown in Table 8. For $e_i \in \mathcal{E}$, the answer to the attribute question Q is formulated as $a_i = e_i(P_a, I, T, Q)$, where P_a is the answer guideline shown in Table 8. After all the experts have generated answers, an additional judge checks the consistency across all answers and the original attribute value. Since experts generate answers in varied forms, they might use diverse phrases to convey the same meaning. We adopt Mixtral $8 \times 7\text{B}$ (Jiang et al., 2024), a powerful large language model with a mixture of experts structure(Fedus et al., 2021), to evaluate the original attribute value by assigning a score s from the experts as shown in Equation 1.

$$s = \sum_{e_i \in \mathcal{E}} \frac{\text{Mixtral}(A_n, A_v, e_i)}{|\mathcal{E}|} \quad (1)$$

Here, $\text{Mixtral}(\cdot, \cdot, \cdot)$ is an indicator function checking whether expert answers are equivalent to the reference attribute value. The adopted prompt is displayed in Table 8. An attribute pair is deemed ha if the score is below a threshold ϵ . Practically, ϵ is set to 0.6, meaning a pair remains only when at least three experts agree with the reference attribute value. Table 2 shows the cleaned data statistics. To illustrate the training set composition, we divided over 4k leaf categories into eight primary domains, selecting the most common attributes for each and displaying them in Figure 3. The size of the segments in the pie chart represents the proportion of each domain within the entire dataset.

3.3 Product Understanding Tasks Formulation

In considering the product listing procedures within actual production environments, we have rigorously designed five tasks aimed at optimizing the efficiency of the overall production process. **(1) Caption Generation (CG):** The task requires the model, given an image of a product, to generate a caption that encapsulates key information about the product. **(2) Product Category Multiple-Choice Question (CMC):** Here, the model must select the most appropriate category from a list of options, based on the product’s image and caption. The options are derived from a category taxonomy, sourced from AliExpress, with at most nine sibling categories sampled to form the choices. **(3)**

Tasks	Num of samples
CG	5,000
CC	960
AI	6,031
AC	5,032
CMC	4,967

Table 3: The statistics of the PUMBENCH.

Attribute Inference (AI): This task involves the model inferring the value of an attribute from the image and caption, based on a provided attribute name. For attributes that are challenging to determine, the model should also reject responding. To achieve this, filtered attributes are reused and their values are designated as 'Unknown'. Building upon these foundational tasks, we developed two advanced tasks. **(4) Caption Completion (CC):** As new attributes are introduced, the model must complete the existing caption to include all necessary keywords for display. For training samples, we eliminate all keywords listed in the attributes from the original captions. **(5) Attribute Correction (AC):** The model’s task is to identify and correct discrepancies between attribute values provided by the seller and other existing information about the product. In case of an error, the model should supply the correct attribute value. For practical purposes, the original value is replaced with a random one. Approximately 30 instructions and 20 response templates were manually designed for each task to ensure diversity. Using a conversation format akin to Qwen-VL-Chat (Bai et al., 2023), specific values are contained within `<>` to facilitate extraction in real scenarios. Table 4 offers several examples of all the tasks, elucidating the details of these five tasks.

4 Benchmarking on Product Understanding

4.1 Implementation details and baselines

Implementation details. We choose Qwen-VL-Chat as our base model and train with LoRA (Hu et al., 2022), a parameter-efficient finetuning method for 3 epochs with batch size 144. The LoRA rank and alpha are 128 and 16 respectively. We employ AdamW (Loshchilov and Hutter, 2017) as the optimizer. The learning rate has a linear warm-up from $1e-8$ to $1e-5$, followed by a cosine-decay from $1e-5$ to 0. The model is trained with 8 Nvidia A100 (80G) GPUs for about 24 hours.

Baselines. We employ InstructBLIP(Dai et al.,



Figure 3: Most common attribute names and proportion of 8 primary domains.

2024), LLaVA-1.5(Liu et al., 2023c), mPlug-Owl2(Ye et al., 2023), MiniGPT-4(Zhu et al., 2023a), Qwen-VL-Chat(Bai et al., 2023) and GPT-4V(Achiam et al., 2023) to be the compared LVLM baselines. Besides, we trained a non-LVLM model based on CLIP(ViT-B/32)(Radford et al., 2021) to perform non-generative tasks for comparison. For both hallucination detection and evaluation on PUMBENCH of all the compared methods, we set temperature and top_p to 0.9 and 0.2 respectively with zero-shot prompting. For GPT-4V, we follow its default options, prompting it with both zero-shot and one-shot settings. We do not use one-shot prompting for open-source models because they are easily hampered by the demonstration given. The further details and the all the prompts used are shown in the Appendix A.2 and Appendix A.1 respectively.

4.2 Datasets and metrics

PUMBENCH. We construct PUMBENCH to evaluate the capabilities of product understanding of PUMGPT and the baselines. We additionally collected 1.5k items from three domains(Home, Electronics, and Clothing) and employed 2 PhD students to clean the attributes to construct the AI and AC test set according to their common sense and construct other task benchmarks as we built the training set. The statistics of PUMBENCH are shown in Table 3.

Metrics. Due to the different output formats and diverse representations of the baselines, we employ

	User	Assistant
CG	What's a suitable caption for this product that would be eye-catching in the image?	A catchy caption is <PROLY New Fashion Women Headband Wide Side Big Bowknot Hairband Classic Casual Turban Girls Headwear Hair Accessories>.
CC	Please incorporate these characteristics: [<[Brand Name, SoundPEATS]>, <[Communication, True Wireless]>, ...], into a revised version of this caption: <Clear Transparent Series Earbuds with Dual ENC Mic, Gaming Mode, App Support, 40H>.	Reflecting the product's attributes, a revised caption would be <SoundPEATS Clear Transparent Series Bluetooth 5.3 True Wireless Earbuds with Dual ENC Mic, Gaming Mode, App Support, 40H>.
AI	What is the attribute <Gender>? If it's indistinct, please reply with 'Unknown'.	The attribute value might be <WOMEN>.
AC	Does the attribute <Material> correctly be <Silk>? Please advise on the accurate value if it's inconsistent.	No, the product's <Material> attribute is <Metal>.
CMC	Pick the most suitable category for the product: 1:<Hair Ties>,2:<Headband>,3:<Hair Claw>,....	It best fits into category <2: Headband>.

Table 4: Examples of each task in the training set. Here we omit the image input.

Tasks	Home	Electronics	Clothing
InstBLIP	10.20	7.17	3.80
LLaVA	22.71	25.26	21.57
Mini	8.75	6.42	3.23
Owl2	20.00	18.85	19.24
Qwen-VL	14.17	25.01	17.83
GPT-4V	29.79	36.04	22.33
GPT-4V†	41.46	45.41	37.18
PUMGPT	32.91	35.49	78.26

Table 5: Domain-level results on attribute inference task.

the Mixtral 8×7B(Jiang et al., 2024) to serve as the answer equivalence judge to determine the accuracy of the attribute-related tasks. For CG and CC tasks, we adopt Bleu₁(Papineni et al., 2002), ROUGE_L(Lin, 2004) and CIDEr(Vedantam et al., 2014) metrics. Besides, we use recall as an additional metric to evaluate the CC task. We utilize accuracy (acc), F1, precision(prec), and recall(rec) to assess the attribution correction task and only accuracy on the CMC task. All reported results are the averages of three separate runs.

5 Experimental Results

5.1 Main Results on PUMBENCH

Table 6 elucidates the comparative performance of PUMGPT and other methodologies on PUMBENCH. Overall, PUMGPT demonstrates superior efficacy across various tasks. Specifically, in the two caption-centric tasks, PUMGPT excels in generating captions by distilling key characteristics from images. This proficiency translates into markedly higher scores on the caption-related metrics, which evaluate the recall and utilization of specific keywords. In the CC task, aided by a base caption, PUMGPT achieves higher performance in caption-related metrics. However, while

GPT-4V successfully recalls nearly all keywords, PUMGPT achieves a recall rate of only 70%. This discrepancy occurs because GPT-4V(zero/one-shot setting) formulates the completed caption from most attribute values in the reference list rather than amending the original title, resulting the lower scores in caption-related metrics.

Regarding the AI tasks, PUMGPT significantly surpasses open-source models and GPT-4V. Notably, for the attribute inference task, PUMGPT exceeds the performance of GPT-4V† by a margin of about twenty percentage points, highlighting the difficulties that advanced commercial models encounter with complex product understanding tasks that demand specialized domain knowledge, even after being presented with a demonstration. Furthermore, due to the stringent regulations, GPT-4V fails to address some test samples involving prohibited topics. In the AC task, PUMGPT maintains an F1 score exceeding 90%, while other models exhibit relatively weaker performance. Many open-source models falter in adhering to the provided instructions, thereby failing to furnish accurate values despite identifying erroneous attributes. Only MiniGPT-4 and GPT-4V can provide corrections, albeit still trailing PUMGPT. Even after fine-tuning, the non-LVLM model CLIP only performs slightly better than some open-source models on this task, while still falling far behind the performance of our model. Additionally, it cannot generate corrected answers as a classifier, indicating that LVLMs have an advantage in tasks requiring both generation and discrimination.

In the CMC task, PUMGPT and GPT-4V exhibited comparable performance, significantly exceeding that of other open-source models and smaller models fine-tuned for this task. Since this task es-

Tasks	InstBLIP	LLaVA	Mini	Owl2	Qwen-VL	GPT-4V	GPT-4V†	CLIP	PumGPT	
CG	Bleu ₁	0.094	0.069	0.086	0.087	0.153	0.102	<u>0.243</u>	-	0.383
	ROUGE _L	0.120	0.073	0.080	0.092	0.148	0.110	<u>0.185</u>	-	0.286
	CIDEr	0.157	0.089	0.181	0.171	0.295	0.128	<u>0.521</u>	-	0.987
CC	Bleu ₁	0.364	0.417	0.538	0.393	0.556	0.442	<u>0.580</u>	-	0.934
	ROUGE _L	0.499	0.379	<u>0.745</u>	0.375	0.480	0.337	0.513	-	0.937
	CIDEr	3.453	1.685	<u>4.410</u>	1.508	2.492	1.281	2.531	-	8.595
AI	Rec(%)	2.86	22.71	10.32	39.74	61.86	92.09	<u>90.39</u>	-	70.63
	Acc(%)	5.45	22.90	4.73	19.25	19.89	26.98	<u>40.24</u>	-	60.70
	F1(%)	67.17	61.35	38.68	60.68	78.82	71.38	<u>80.09</u>	68.79	93.14
AC	Prec(%)	50.60	56.38	65.50	61.74	76.00	81.11	<u>83.65</u>	60.39	90.34
	Rec (%)	99.88	67.30	27.44	59.65	81.87	63.74	76.83	79.90	<u>96.12</u>
	CAcc(%)	0.98	0.48	41.16	1.25	0.39	50.01	<u>54.14</u>	-	60.52
CMC	Acc(%)	25.21	31.45	33.56	62.04	48.44	82.55	83.02	39.76	<u>82.57</u>

Table 6: The experimental results on PUMBENCH, where CAcc is the accuracy of the attribute correction. We abbreviate the models for better vision effect, where InstBLIP is for InstructBLIP, Mini for MiniGPT-4, Owl2 for mPlug-Owl2, Qwen-VL for Qwen-VL-Chat. We report the results * 100% for all the metrics except for the Bleu₁, ROUGE_L, and CIDEr. † means the model is equipped with one-shot prompting.

essentially involves reasoning that does not require product knowledge, it is evident that GPT-4V performs nearly equally well in both zero-shot and one-shot settings, indicating that it already possesses strong multi-choice reasoning capabilities. Despite our model being trained, it did not significantly surpass GPT-4V’s performance, which shows that there is still room for further improvement in this task.

5.2 Domain-level Results on Attribute Inference

The attribute inference task test set is divided into three categories: Home, Electronics, and Clothing. Home and Electronics consist of standardized goods, where most attributes and values are pre-defined and can be directly extracted from titles and specifications. In contrast, Clothing represents non-standardized goods, with attributes that may be vendor-specific and open to interpretation. For example, a garment’s style could be labeled both "commute" and "casual," requiring models to learn vendor-specific styles during training, focusing on specific distributions.

Table 5 shows the performance of each method. Overall, PUMGPT outperforms other models, but in the Home and Electronics domains, it is less effective than GPT-4V with one-shot prompting, despite surpassing GPT-4V with zero-shot prompting. Error analysis revealed that some test cases involve extracting spans (e.g., model numbers), which PUMGPT struggles with. In these cases, GPT-4V with one-shot prompting can treat it as an NER (extractive) task, yielding better results. A

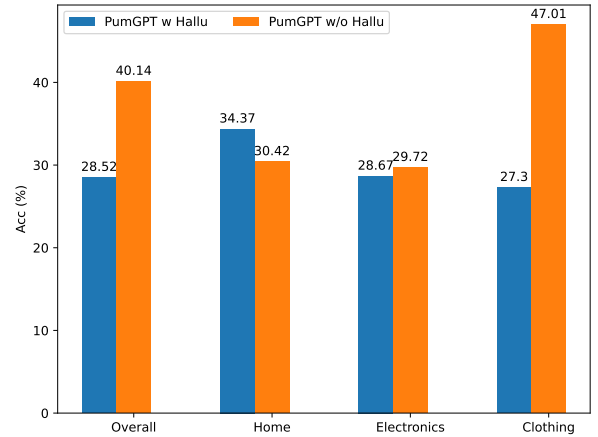


Figure 4: Ablation on hallucination filtering. Here we report the accuracy of the attribution inference task, where w Hallu means it was trained on the hallucination dataset and w/o Hallu means was trained on the hallucination-free dataset.

hard case is shown in Table 12 in the appendix, where most models fail. For non-standardized goods, PUMGPT excels at attribute inference by effectively learning from product data and capturing vendor-specific descriptions. In contrast, models without such training only reflect their pre-training distributions, performing inadequately for real-world applications.

5.3 Ablation on Hallucination Filtering

In the attribute inference task, PUMGPT achieved more than twice the accuracy of GPT-4V, prompting an investigation into whether this improvement was due to its handling of hallucinations. We conducted an ablation study on hallucination process-

Models	F1	Prec	Rec
InstBLIP	0	0	0
LLaVA	17.67	20.95	15.27
Mini	0.75	4.44	0.41
Owl2	11.11	8.73	15.27
Qwen-VL	12.66	8.79	22.60
GPT-4V	29.69	19.33	64.01
GPT-4V†	47.33	47.66	47.00
PUMGPT	47.18	55.22	41.12

Table 7: The evaluation on the rejection ability of all the compared methods.

ing, extracting a 600k subset from the original 663k dataset. For the hallucination dataset, up to eight attributes per product were randomly sampled for training. For the hallucination-free dataset, the methods in Section 3.2 were applied, limiting the number of attributes (including "unknown" ones) to eight. Both models were trained for two epochs with identical parameters.

Figure 4 shows that PUMGPT without hallucination data (w/o Hallu) demonstrated significant performance improvement. The accuracy was categorized into three primary groups as in Section 5.2. In the standardized categories, model performance was similar. In the Home category, PUMGPT with hallucination data (w Hallu) outperformed its counterpart by about four percentage points, as it learned more attributes. However, in the Clothing category, PUMGPT w/o Hallu outperformed the other by nearly 20 percentage points. The Clothing category mostly involves non-standardized items with subjective attributes, where training with hallucinated data can lead to overly imaginative but inaccurate responses. In contrast, the hallucination-free dataset reduced such extrapolations, yielding more accurate predictions. Thus, hallucination processing is crucial for model training.

5.4 Evaluation on Rejection Ability

Large language models are praised for their text completion capabilities but may generate incorrect information due to excessive associative reasoning. In practical applications, a model should avoid answering when faced with nonexistent or ambiguous attributes, instead of providing plausible but incorrect responses.

As shown in Table 7, these metrics are derived by treating rejection as a binary classification in the attribute inference task. Open-source models like InstructBLIP and MiniGPT-4 tend to provide actual

values rather than rejecting, leading to lower recall. Specifically, InstructBLIP never refuses, yielding zero across all metrics. In contrast, GPT-4V attempts more refusals with zero-shot prompting but struggles with precision due to conservative rules. With one demonstration, GPT-4V improves its ability to reject or answer, increasing precision and overall accuracy compared to the zero-shot setting. While our model’s recall is lower than GPT-4V, it significantly outperforms in precision, highlighting the effectiveness of training with "unknown" attributes. Further improvement of rejection capabilities may require preference learning algorithms like PPO (Schulman et al., 2017) and DPO (Rafailov et al., 2023).

5.5 Evaluation on OOD Attribute Data

We conducted a small-scale experiment on ImplicitAVE(Zou et al., 2024), an out-of-domain (OOD) dataset. Results show that PUMGPT outperforms the base model, though differences in data distribution across e-commerce platforms, especially in attribute granularity and label space. To improve performance across platforms, incorporating additional in-distribution data for continued training may be effective. Details are in Appendix A.3.

5.6 Case Study

We also showed two cases in the Appendix A.4 to delve into the PUMGPT’s advantages and disadvantages as a further analysis of the domain-level results on the AI task.

6 Conclusion

In this work, we introduce PUMGPT, the pioneering Large Vision Language Model (LVLM) for e-commerce product understanding. We amassed over one million product entries and employed a multi-expert collaborative hallucination handling framework to eliminate mislabeled attributes or those not inferable from text and images. We devised five product understanding tasks aligned with actual product listing processes, resulting in a dataset of approximately 663k entries to train PUMGPT. We also developed PUMBENCH to assess the performance of PUMGPT and other LVLMs in product understanding. Experimental results reveal that PUMGPT outperforms non-LVLM baseline, general-purpose LVLMs, such as GPT-4V. Future work will expand task variety and improve data quality to enhance model performance further.

Limitations

Although PUMGPT demonstrated superior performance in evaluations, it still has some limitations. (1) in the CMC task, PUMGPT’s performance did not significantly surpass GPT-4V. Additionally, there is a considerable accuracy gap between standardized product attribute inference tasks and non-standardized product tasks. Introducing more trainable parameters or applying preference learning algorithms to specifically enhance these tasks is necessary. (2) we designed only five product understanding tasks for training, which resulted in a weaker generalization ability of the model. This limitation makes it challenging to extend to other advanced product understanding tasks, such as identifying identical products and generating product descriptions. Consequently, the model’s capacity to leverage the full potential of large language models is still insufficient. To address these limitations, it is necessary to introduce a greater variety and diversity of task data. This should include not only task-specific data but also general instruction data to improve the model’s generalization capability. (3) Lack further quality test on the hallucination filtering. To further demonstrate the effectiveness of our hallucination detection framework, human experts evaluation is needed.

References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report.
- Aman Ahuja, Nikhil Rao, Sumeet Katariya, Karthik Subbian, and Chandan K. Reddy. 2020. [Language-agnostic representation learning for product search on e-commerce platforms](#). In *Proceedings of the 13th International Conference on Web Search and Data Mining, WSDM ’20*, page 7–15, New York, NY, USA. Association for Computing Machinery.
- Qingyao Ai, Yongfeng Zhang, Keping Bi, Xu Chen, and W Bruce Croft. 2017. Learning a hierarchical embedding model for personalized product search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 645–654.
- Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. 2022. [Flamingo: a visual language model for few-shot learning](#). *ArXiv*, abs/2204.14198.
- Birkan Atıcı and Sevinç İlhan Omurca. 2021. Generating classified ad product image titles with image captioning. In *Trends in Data Engineering Methods for Intelligent Systems: Proceedings of the International Conference on Artificial Intelligence and Applied Mathematics in Engineering (ICAIAME 2020)*, pages 211–219. Springer.
- Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. [Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond](#). *arXiv preprint arXiv:2308.12966*, arXiv:2308.12966.
- Christopher Bonnett. 2016. Classifying e-commerce products based on images and text. *Adventures in Machine Learning*.
- Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. [Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality](#).
- Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. 2024. Instructblip: Towards general-purpose vision-language models with instruction tuning. *Advances in Neural Information Processing Systems*, 36.

645	Xiao Dong, Xunlin Zhan, Yangxin Wu, Yunchao Wei,	on product aspects. In <i>Proceedings of the 14th ACM</i>	702
646	Xiaoyong Wei, Minlong Lu, and Xiaodan Liang.	<i>International Conference on Web Search and Data</i>	703
647	2021. M5product: A multi-modal pretraining bench-	<i>Mining</i> , pages 967–975.	704
648	mark for e-commercial product downstream tasks.		
649	<i>ArXiv</i> , abs/2109.04275.		
650	Danny Driess, F. Xia, Mehdi S. M. Sajjadi, Corey	Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H.	705
651	Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan	Hoi. 2023a. Blip-2: Bootstrapping language-image	706
652	Wahid, Jonathan Tompson, Quan Ho Vuong, Tianhe	pre-training with frozen image encoders and large	707
653	Yu, Wenlong Huang, Yevgen Chebotar, Pierre Ser-	language models . In <i>International Conference on</i>	708
654	manet, Daniel Duckworth, Sergey Levine, Vincent	<i>Machine Learning</i> .	709
655	Vanhoucke, Karol Hausman, Marc Toussaint, Klaus		
656	Greff, Andy Zeng, Igor Mordatch, and Peter R. Flo-	Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang,	710
657	rence. 2023. Palm-e: An embodied multimodal lan-	Wayne Xin Zhao, and Ji rong Wen. 2023b. Eval-	711
658	guage model . In <i>International Conference on Ma-</i>	uating object hallucination in large vision-language	712
659	<i>chine Learning</i> .	models . In <i>Conference on Empirical Methods in</i>	713
		<i>Natural Language Processing</i> .	714
660	William Fedus, Barret Zoph, and Noam M. Shazeer.	Chin-Yew Lin. 2004. Rouge: A package for automatic	715
661	2021. Switch transformers: Scaling to trillion pa-	evaluation of summaries . In <i>Annual Meeting of the</i>	716
662	parameter models with simple and efficient sparsity . <i>J.</i>	<i>Association for Computational Linguistics</i> .	717
663	<i>Mach. Learn. Res.</i> , 23:120:1–120:39.		
664	Anish Gunjal, Jihan Yin, and Erhan Bas. 2023. De-	Rongmei Lin, Xiang He, Jie Feng, Nasser Zalmout, Yan	718
665	tecting and preventing hallucinations in large vision	Liang, Li Xiong, and Xin Luna Dong. 2021. Pam:	719
666	language models . In <i>AAAI Conference on Artificial</i>	understanding product images in cross product cate-	720
667	<i>Intelligence</i> .	gory attribute extraction. In <i>Proceedings of the 27th</i>	721
		<i>ACM SIGKDD Conference on Knowledge Discovery</i>	722
		<i>& Data Mining</i> , pages 3262–3270.	723
668	Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan	Fan Liu, Delong Chen, Xiaoyu Du, Ruizhuo Gao, and	724
669	Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and	Feng Xu. 2023a. Mep-3m: A large-scale multi-	725
670	Weizhu Chen. 2022. LoRA: Low-rank adaptation of	modal e-commerce product dataset . <i>Pattern Recog-</i>	726
671	large language models . In <i>International Conference</i>	<i>nit.</i> , 140:109519.	727
672	<i>on Learning Representations</i> .		
673	Qidong Huang, Xiao wen Dong, Pan Zhang, Bin Wang,	Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser	728
674	Conghui He, Jiaqi Wang, Dahua Lin, Weiming	Yacoob, and Lijuan Wang. 2023b. Mitigating hal-	729
675	Zhang, and Neng H. Yu. 2023a. Opera: Alleviating	lucination in large multi-modal models via robust	730
676	hallucination in multi-modal large language models	instruction tuning . In <i>International Conference on</i>	731
677	via over-trust penalty and retrospection-allocation .	<i>Learning Representations</i> .	732
678	<i>ArXiv</i> , abs/2311.17911.		
679	Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao,	Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae	733
680	Saksham Singhal, Shuming Ma, Tengchao Lv, Lei	Lee. 2023c. Visual instruction tuning.	734
681	Cui, Owais Khan Mohammed, Qiang Liu, Kriti Ag-		
682	garwal, Zewen Chi, Johan Bjorck, Vishrav Chaud-	Ilya Loshchilov and Frank Hutter. 2017. Decoupled	735
683	hary, Subhojit Som, Xia Song, and Furu Wei. 2023b.	weight decay regularization . In <i>International Confer-</i>	736
684	Language is not all you need: Aligning perception	<i>ence on Learning Representations</i> .	737
685	with language models . <i>ArXiv</i> , abs/2302.14045.		
686	Albert Q. Jiang, Alexandre Sablayrolles, Antoine	Kishore Papineni, Salim Roukos, Todd Ward, and Wei-	738
687	Roux, Arthur Mensch, Blanche Savary, Chris Bam-	Jing Zhu. 2002. Bleu: a method for automatic evalu-	739
688	ford, Devendra Singh Chaplot, Diego de Las Casas,	ation of machine translation . In <i>Annual Meeting of</i>	740
689	Emma Bou Hanna, Florian Bressand, Gianna	<i>the Association for Computational Linguistics</i> .	741
690	Lengyel, Guillaume Bour, Guillaume Lample,		
691	L'elio Renard Lavaud, Lucile Saulnier, Marie-	Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya	742
692	Anne Lachaux, Pierre Stock, Sandeep Subramanian,	Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sas-	743
693	Sophia Yang, Szymon Antoniak, Teven Le Scao,	try, Amanda Askell, Pamela Mishkin, Jack Clark,	744
694	Théophile Gervet, Thibaut Lavril, Thomas Wang,	Gretchen Krueger, and Ilya Sutskever. 2021. Learn-	745
695	Timothée Lacroix, and William El Sayed. 2024. Mix-	ing transferable visual models from natural language	746
696	tral of experts . <i>ArXiv</i> , abs/2401.04088.	supervision . In <i>Proceedings of the 38th International</i>	747
697		<i>Conference on Machine Learning</i> , volume 139 of	748
698	Diederik P. Kingma and Jimmy Ba. 2014. Adam:	<i>Proceedings of Machine Learning Research</i> , pages	749
699	A method for stochastic optimization . <i>CoRR</i> ,	8748–8763. PMLR.	750
	abs/1412.6980.		
700	Trung-Hoang Le and Hady W Lauw. 2021. Explain-	Rafael Rafailov, Archit Sharma, Eric Mitchell, Christo-	751
701	able recommendation with comparative constraints	pher D Manning, Stefano Ermon, and Chelsea Finn.	752
		2023. Direct preference optimization: Your language	753
		model is secretly a reward model . In <i>Thirty-seventh</i>	754
		<i>Conference on Neural Information Processing Sys-</i>	755
		<i>tems</i> .	756

757	John Schulman, Filip Wolski, Prafulla Dhariwal, Alec	Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen	815
758	Radford, and Oleg Klimov. 2017. Proximal policy	Hu, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and	816
759	optimization algorithms . <i>ArXiv</i> , abs/1707.06347.	Jingren Zhou. 2023. mplug-owl2: Revolutionizing	817
760	Keiji Shinzato, Naoki Yoshinaga, Yandi Xia, and Wei-	multi-modal large language model with modality col-	818
761	Te Chen. 2022. Simple and effective knowledge-	laboration . <i>Preprint</i> , arXiv:2311.04257.	819
762	driven query expansion for qa-based product attribute	Xunlin Zhan, Yangxin Wu, Xiao Dong, Yunchao Wei,	820
763	extraction. In <i>Proceedings of the 60th Annual Meet-</i>	Minlong Lu, Yichi Zhang, Hang Xu, and Xiaodan	821
764	<i>ing of the Association for Computational Linguistics</i>	Liang. 2021. Product1m: Towards weakly super-	822
765	<i>(Volume 2: Short Papers)</i> , pages 227–234.	vised instance-level product retrieval via cross-modal	823
766	Changfeng Sun, Han Liu, Meng Liu, Zhaochun Ren,	pretraining . <i>2021 IEEE/CVF International Con-</i>	824
767	Tian Gan, and Liqiang Nie. 2020. Lara: Attribute-	<i>ference on Computer Vision (ICCV)</i> , pages 11762–	825
768	to-feature adversarial learning for new-item recom-	11771.	826
769	mendation. In <i>Proceedings of the 13th international</i>	Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu,	827
770	<i>conference on web search and data mining</i> , pages	Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and	828
771	582–590.	Yu Jiao Qiao. 2023a. Llama-adapter: Efficient fine-	829
772	Haoqin Tu, Chenhang Cui, Zijun Wang, Yiyang Zhou,	tuning of language models with zero-init attention .	830
773	Bingchen Zhao, Junlin Han, Wangchunshu Zhou,	<i>ArXiv</i> , abs/2303.16199.	831
774	Huaxiu Yao, and Cihang Xie. 2023. How many uni-	Yupeng Zhang, Shensi Wang, Peiguang Li, Guanting	832
775	corns are in this image? a safety evaluation bench-	Dong, Sirui Wang, Yunsen Xian, Zhoujun Li, and	833
776	mark for vision llms . <i>ArXiv</i> , abs/2311.16101.	Hongzhi Zhang. 2023b. Pay attention to implicit	834
777	Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi	attribute values: a multi-modal generative frame-	835
778	Parikh. 2014. Cider: Consensus-based image descrip-	work for ave task. In <i>Findings of the Association</i>	836
779	tion evaluation . <i>2015 IEEE Conference on Computer</i>	<i>for Computational Linguistics: ACL 2023</i> , pages	837
780	<i>Vision and Pattern Recognition (CVPR)</i> , pages 4566–	13139–13151.	838
781	4575.	Zhiyuan Zhao, Bin Wang, Linke Ouyang, Xiao wen	839
782	Qifan Wang, Li Yang, Bhargav Kanagal, Sumit Sanghai,	Dong, Jiaqi Wang, and Conghui He. 2023. Be-	840
783	D Sivakumar, Bin Shu, Zac Yu, and Jon Elsas. 2020.	yond hallucinations: Enhancing lvlms through	841
784	Learning to extract attribute value from product via	hallucination-aware direct preference optimization .	842
785	question answering: A multi-task approach. In <i>Pro-</i>	<i>ArXiv</i> , abs/2311.16839.	843
786	<i>ceedings of the 26th ACM SIGKDD international</i>	Guineng Zheng, Subhabrata Mukherjee, Xin Luna	844
787	<i>conference on knowledge discovery & data mining</i> ,	Dong, and Feifei Li. 2018. Opentag: Open attribute	845
788	pages 47–55.	value extraction from product profiles. In <i>Proceed-</i>	846
789	Wenyi Xiao, Ziwei Huang, Leilei Gan, Wanggui He,	<i>ings of the 24th ACM SIGKDD international confer-</i>	847
790	Haoyuan Li, Zhelun Yu, Hao Jiang, Fei Wu, and Lin-	<i>ence on knowledge discovery & data mining</i> , pages	848
791	chao Zhu. 2024. Detecting and mitigating hallucina-	1049–1058.	849
792	tion in large vision language models via fine-grained	Yiyang Zhou, Chenhang Cui, Rafael Rafailov, Chelsea	850
793	ai feedback . <i>ArXiv</i> , abs/2404.14233.	Finn, and Huaxiu Yao. 2024. Aligning modalities	851
794	Huimin Xu, Wenting Wang, Xinnian Mao, Xinyu Jiang,	in vision large language models via preference fine-	852
795	and Man Lan. 2019. Scaling up open tagging from	tuning . <i>ArXiv</i> , abs/2402.11411.	853
796	tens to thousands: Comprehension empowered at-	Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and	854
797	tribute value extraction from product title. In <i>Pro-</i>	Mohamed Elhoseiny. 2023a. Minigpt-4: Enhancing	855
798	<i>ceedings of the 57th Annual Meeting of the Asso-</i>	vision-language understanding with advanced large	856
799	<i>ciation for Computational Linguistics</i> , pages 5214–	language models. <i>arXiv preprint arXiv:2304.10592</i> .	857
800	5223.	Tiangang Zhu, Yue Wang, Haoran Li, Youzheng Wu,	858
801	Jun Yan, Nasser Zalmout, Yan Liang, Christan Grant,	Xiaodong He, and Bowen Zhou. 2020. Multimodal	859
802	Xiang Ren, and Xin Luna Dong. 2021. Adatag:	joint attribute prediction and value extraction for e-	860
803	Multi-attribute value extraction from product profiles	commerce product. In <i>Proceedings of the 2020 Con-</i>	861
804	with adaptive decoding. In <i>Proceedings of the 59th</i>	<i>ference on Empirical Methods in Natural Language</i>	862
805	<i>Annual Meeting of the Association for Computational</i>	<i>Processing (EMNLP)</i> , pages 2129–2139.	863
806	<i>Linguistics and the 11th International Joint Confer-</i>	Zihao Zhu, Mingda Zhang, Shaokui Wei, Bing Wu, and	864
807	<i>ence on Natural Language Processing (Volume 1:</i>	Baoyuan Wu. 2023b. Vdc: Versatile data cleanser	865
808	<i>Long Papers)</i> , pages 4694–4705.	based on visual-linguistic inconsistency by multi-	866
809	Li Yang, Qifan Wang, Zac Yu, Anand Kulkarni, Sumit	modal large language models . In <i>International Con-</i>	867
810	Sanghai, Bin Shu, Jon Elsas, and Bhargav Kanagal.	<i>ference on Learning Representations</i> .	868
811	2022. Mave: A product dataset for multi-source	Zihao Zhu, Mingda Zhang, Shaokui Wei, Bingzhe Wu,	869
812	attribute value extraction. In <i>Proceedings of the fif-</i>	and Baoyuan Wu. 2024. Vdc: Versatile data cleanser	870
813	<i>teenth ACM international conference on web search</i>		
814	<i>and data mining</i> , pages 1256–1265.		

based on visual-linguistic inconsistency by multi-modal large language models. In *The Twelfth International Conference on Learning Representations*.

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

A Appendix

A.1 Prompts

Here we provide all the prompts used for generating attribute questions, checking equivalent attribute values, and benchmarking in table 8. For all the models during inference, we use the same prompts shown in Table 8. The one-shot prompt is similar but prepend a demonstration before the real question.

A.2 Model Details

Table 9 shows the details of the model we compared and other generation configs. For GPT-4V, we call the Azure API and its version is '2023-12-01-preview'. For CLIP trained on AC task, we fuse the features of the product image, title, and attribute assertion and feed them into a classification head to predict a score. The threshold for inference is 0.5. The attribute assertion template is 'The [PLACEHOLDER] attribute of the product in the image is [PLACEHOLDER]'. For the CMC task, we train the CLIP following its original contrastive learning paradigm where we contrast the category feature with a fused feature of product image and title. The template used is 'The finest category of the product in the image is [PLACEHOLDER]'. For both models, we set the batch size and lr at 128 and 5e-5. We choose the Adam(Kingma and Ba, 2014) optimizer and set the betas to be (0.9, 0.98) and eps to be 1e-6. The CLIP model was trained on one NVIDIA A10 GPU for 2 epochs with full parameters. All the experiments were conducted under a torch2.01+cu118 environment. Note that all the compared methods were prompted without the special token <>.

A.3 Evaluation on OOD Attribute Data

We evaluated on ImplicitAVE(Zou et al., 2024), a small-scale attribute inference dataset with approximately 1.6k samples in the test set, encompassing 25 attributes and 158 attribute values. Although the attribute scale is much smaller than our training set, these attributes were originally derived from automatically annotated Amazon review datasets, resulting in a vastly different distribution compared to ours. The original ImplicitAVE dataset only required the model to select an attribute value from multiple choices, resembling the CMC task in our paper. However, our approach involves the free generation of attribute values, making direct comparison with the performance metrics of other mod-

els in the original ImplicitAVE paper infeasible. Due to evaluation cost constraints(human evaluation), we only tested PUMGPT and an untrained Qwen-VL-Chat on this dataset for the task of free attribute inference. We asked two master's students to independently evaluate the results based on product images, attribute names, and reference attribute values. Table 10 shows the results:

After training, our model exhibits significant performance improvements in OOD data compared to the base model. We analyzed some of the errors made by PUMGPT: 1) Our model tends to predict unknown attribute values. Consequently, it may refuse to respond to certain queries that require domain-specific knowledge not encountered during training (e.g., candy variety: Licorice). 2) The set of values for attribute names can differ, such as numerical specifications versus descriptive specifications for Shaft Height. 3) This dataset often combines multiple finer-grained attributes into a single coarser-grained attribute, such as merging length into style. These differences highlight the distributional discrepancies between ImplicitAVE and our training set. Despite the performance degradation caused by these distributional differences, our model still manages to infer correct attribute values in most cases which is more effective than the untrained baseline model. In conclusion, there are significant differences in data distribution across various e-commerce platforms, particularly in terms of attribute granularity and label space. To further enhance performance on different e-commerce platforms, leveraging additional in distribution domain data for continued training on PUMGPT may be a viable solution.

A.4 Case Study

We also conducted a case study. Table 11 and Table 12 respectively display the results of all the models for a certain attribute on non-standardized and standardized products, which can also serve as a good and a bad case. For the first case in Table 10, It can be observed that most models are unable to infer results for the non-standardized product. For GPT-4V with the zero-shot setting, it refused to respond possibly due to its conservative rules as we analyzed in experiments and followed our instruction 'respond unknown if you're not sure'. However, once prompted with one demonstration, it can provide a plausible answer. Other open-source models either fail to generate the results or mistakenly output the entire product title while intending to

	Prompt
Question Gen(P_q)	<p>Given the title of a product and a pair of attribute name and value of the product, generate a possible question about the attribute name from which the attribute value can be inferred. The question generated should not contain the attribute value and use a brief name(e.g. just a noun) to refer the product itself.</p> <p>Example: Product name: 4MP 1080P IP Outdoor WiFi Security Camera for Home Surveillance, Waterproof Bullet Cam, HD WiFi Video. Attribute name: Supported Mobile Systems. Attribute value: Android. Question: What is the supported mobile systems of the camera? Product name: [PLACEHOLDER]. Attribute name: [PLACEHOLDER]. Attribute value: [PLACEHOLDER]. Question:</p>
Expert Question Answer(P_a)	<p>The title of the product in the image is [PLACEHOLDER], answer the question as briefly as possible and loyally according to the title and question. Question: [PLACEHOLDER]. Answer:</p>
Answer Check(<i>Mixtral</i>)	<p>Given a certain attribute of a product, you're required to judge whether a candidate attribute value is completely equivalent to the reference attribute value without any ambiguity (consistent keywords and the same number of keywords). Simply respond with "yes" (indicating the two values are equivalent) or "no" (indicating they're not).</p> <p>Attribute name: [PLACEHOLDER]. Reference attribute value: [PLACEHOLDER]. Candidate attribute value: [PLACEHOLDER]. Judgement:</p>
CG	Generate a caption that encapsulates the essence of the product in the image.
CC	Enhance this existing title to make it more appealing for the product shown in the image with these features given: [PLACEHOLDER]. The initial caption is: [PLACEHOLDER]. Just answer the enhanced caption based on the initial caption with necessary attributes.
AI	The caption of the product in the image is [PLACEHOLDER]. Please clarify the attribute [PLACEHOLDER] of the product. Just respond with a simple phrase and respond unknown if you're not sure.
AC	Upon reviewing the product titled [PLACEHOLDER] in the attached image, is the [PLACEHOLDER] attribute correctly [PLACEHOLDER]? Just respond yes or no. If this is incorrect, kindly provide the accurate value.
CMC	The caption of the product in the image is [PLACEHOLDER]. Choose the most fitting category for the product: [PLACEHOLDER]. Just answer the option number that you believe correct.

Table 8: The prompt used for generating attribute questions, checking equivalent attribute values, and benchmarking.

Models	LLM Size	LLM	Beams	Temperature	Top_p
InstBLIP	7B	Vicuna	2	0.9	0.2
LLaVA	7B	LLaMA	2	0.9	0.2
Mini	7B	LLaMA-2	2	0.9	0.2
Owl2	7B	LLaMA-2	2	0.9	0.2
Qwen-VL	7B	Qwen	2	0.9	0.2
GPT-4V	/	/	/	/	/
PUMGPT	7B	Qwen	2	0.9	0.2

Table 9: The details of model size and their base LLMs.

ImplicitAVE	Qwen-VL-Chat	PumGPT
Acc	35.65	58.70

Table 10

express prominent text on the clothes, leading to errors. However, PUMGPT effectively avoided this issue and accurately inferred the correct attribute values.

For the standardized product, the attribute "Model Number" is challenging to determine. Consequently, almost all models performed poorly. Other models directly refused to answer, while PUMGPT attempted to extract a reasonable model number from the title. Despite this effort, it similarly repeated the entire title, as other models did in the previous case. This indicates that PUMGPT still has deficiencies in extracting complex attributes. Addressing this issue may require more difficult samples for training.


Product	Models	Answers
	InstructBLIP	What you're looking for Be Strong And Courageous Pandemicc Political Insurrection Men'S T-Shirt Custom Aldult Teen Unisex Digital Printing Xs-5Xl.
	LLaVA-1.5	Unisex - Youth.
	MiniGPT-4	The image shows a black t-shirt with the words "Be Strong And Courageous".
	mPlug-Owl2	The product is a black t-shirt featuring the phrase "Be Strong and Courageous" and a picture of a fist. It is sold by the brand toulo, available in sizes XS to 5XL.
Caption: Be Strong And Courageous Pandemicc Political Insurrection Men'S T-Shirt Custom Aldult Teen Unisex Digital Printing Xs-5Xl.	Qwen-VL-Chat	The pattern of the product is a simple phrase "Be Strong And Courageous Pandemicc Political Insurrection Men'S T-Shirt Custom Aldult Teen Unisex Digital Printing Xs-5Xl".
	GPT-4V	Unknown.
	GPT-4V†	Text.
	PUMGPT	The attribute value of the product may be <Print>.
Attribute name: Pattern	GT	Print.

Table 11: A case on a non-standardized product, where GT is the reference attribute value

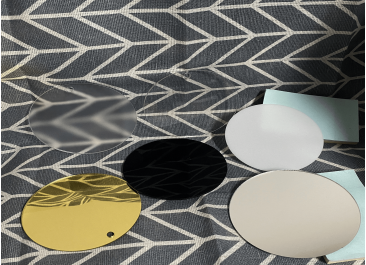
Product	Models	Answers
 <p>Caption: 2/3/4/5/6" Acrylic Circle Sheet Disc Round Board for Wedding Place Card Gift Tag Ornament DIY Craft Art Project Painting Carve.</p> <p>Attribute name: Model Number</p>	InstructBLIP	No answer.
	LLaVA-1.5	Unknown.
	MiniGPT-4	Thank you for providing the image. The product in the image is an acrylic circle sheet disc.
	mPlug-Owl2	The model number of the product is not visible in the image.
	Qwen-VL-Chat	The model number of the product is not mentioned in the not provided.
	GPT-4V	Unknown.
	GPT-4V†	Unknown.
	PUMGPT	The attribute value of the product may be <2/3/4/5/6" Acrylic Circle Sheet Disc Round Board for Wedding Place Card Gift Tag Ornament DIY Craft Art Project Painting Carve>.
	GT	Acrylic Circle Sheet Disc Round Board.

Table 12: A case on a standardized product.