PUMGPT: A Large Vision-Language Model for Product Understanding

Anonymous ACL submission

Abstract

E-commerce platforms benefit from accurate product understanding to enhance user 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 other open-source LVLMs and GPT-4V 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 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 et al., 2020; Ai et al., 2017) on platforms but also facilitates retailers to launch and update their goods with substantial time savings.

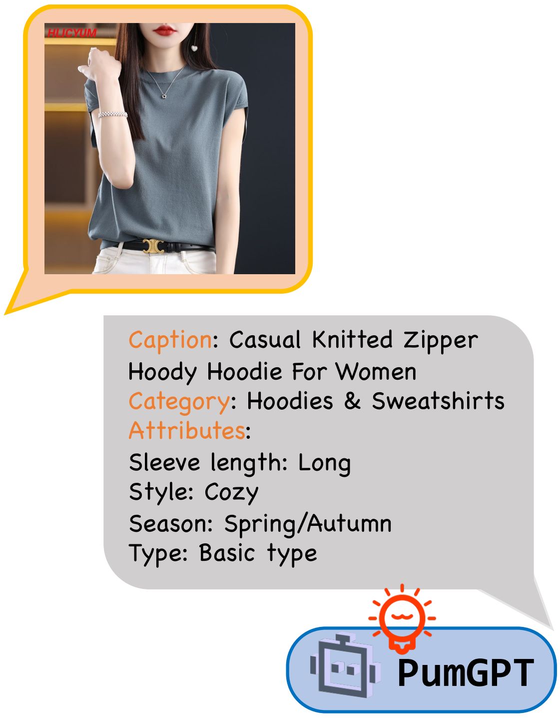


Figure 1: A glimpse on PUMGPT in product understanding.

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., 2023; Liu et al., 2023; 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 platform[1], 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 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 trained on a 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.

[1] https://www.aliexpress.com/

2 Related Works

Vision-Language Models. Recent advancements have shown significant success in leveraging large language models for vision-language tasks. Notable among these, Flamingo(Alayrac et al., 2022) employs a gated cross-attention mechanism to align vision representations with language models. Blip-2(Li et al., 2023) introduces a Q-Former to effectively bridge the gap between visual and textual representations. Moreover, models like Kosmos-1(Huang et al., 2023) and PaLM-E(Driess et al., 2023) achieve alignment between multi-modal and text representations, creating a comprehensive interface for multi-modal input with large language models. GPT-4(Achiam et al., 2023) has demonstrated robust visual reasoning abilities across diverse vision-linguistic tasks. Unlike end-to-end model training, some approaches coordinate multiple models to interpret and respond to multi-modal inputs, exemplified by Visual ChatGPT(Wu et al., 2023), MM-REACT(Yang et al., 2023), and HuggingGPT(Shen et al., 2023). Increasing model sizes raise computational complexity and training data demands, prompting recent studies to explore efficient finetuning methodologies for large vision-language models(Zhu et al., 2023; 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., 2023; 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 models. Product understanding tasks encompass a variety of sub-tasks, with attribute extraction being the most extensively studied. Traditional approaches employ tagging-based models (Zheng et al., 2018; Xu et al., 2019; Yan et al., 2021) or question-answer-based models (Shinzato et al., 2022) to extract attributes from textual product profiles. 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). This fusion of textual and visual data enriches the model's comprehension and extraction capabilities. Besides attribute extraction, other product understanding tasks such as product captioning (Atıcı and İlhan Omurca, 2021) and product classification

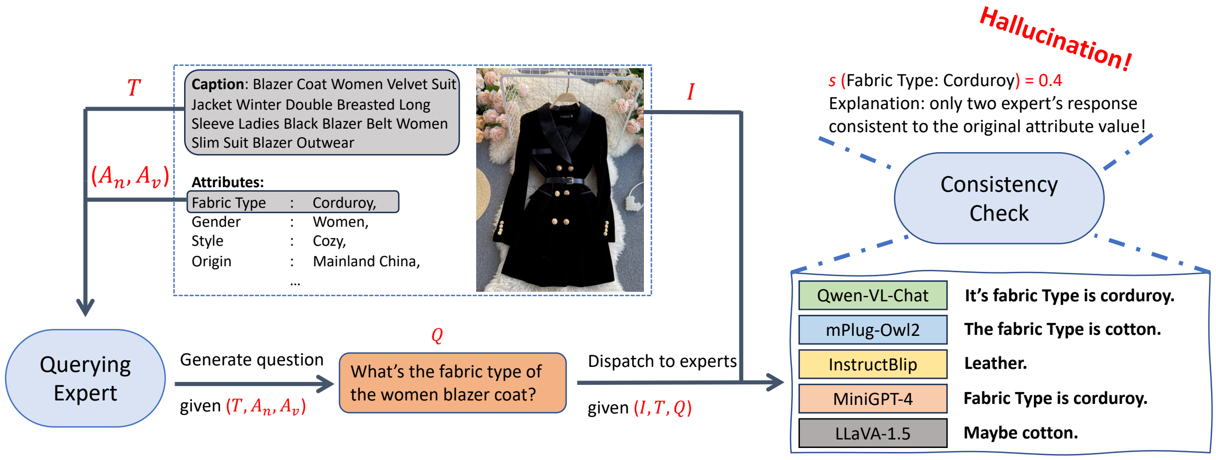


Figure 2: The overview of our proposed hallucination detection framework.

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

(Bonnett, 2016) have also been explored. However, these solutions typically necessitate training separate models for each task. In contrast, our **PUMGPT** integrates all product understanding tasks, significantly improving performance across tasks due to diverse training data and the intrinsic capabilities of **PUMGPT**.

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 from the AliExpress platform. Each product entry contains an image, a caption, the product category, and a set of product attributes. Each attribute consists of an attribute name and a corresponding attribute value. Table 1 demonstrates the statistical results of the 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 categories or attributes, making them unsuitable for training. Additionally, certain attributes might either complement product descriptions and images or conflict with other information sources due to sellers' errors. 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, leading to significant variability in their inference behaviours. Despite these differences, LVLMs can reach consensus on tasks requiring common knowledge or reasoning, while they generate divergent speculations when faced with ambiguous queries. This property can be exploited to detect inconsistencies within product datasets, particularly where attributes misalign with product descriptions and images. By utilizing distinct LVLMs, each with unique knowledge backgrounds, more consistent responses can be generated for accurate attribute values, whereas varied responses indicate mismatched or supplementary information or subjectively valued attributes.

As shown in Figure 2, we selected five LVLMs as experts in hallucination detection: $\mathcal{E} = \{$Qwen-VL-Chat(Bai et al., 2023), MiniGPT-4(Zhu et al., 2023), InstructBLIP(Dai et al., 2024), mPLUG-Owl2(Ye et al., 2023), LLaVA(Liu et al., 2023)$\}$. 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), a querying expert generates questions about A_v. As A_n is not a typed item, the Vicuna-13B(Chiang et al., 2023) querying expert generates a question $Q = Vicuna(P_q, T, A_n, A_v)$ based on the attribute value type. The prompt P_q for generating questions is shown in Table 8.

For $e_i \in \mathcal{E}$, the answer to attribute question Q is formulated as $a_i = e_i(I, T, Q)$. After generating all expert 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 Mistral 8×7B (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.

$$s = \sum_{e_i}^{\mathcal{E}} \frac{Mistral(e_i, A_v)}{|\mathcal{E}|} \quad (1)$$

Here, $Mistral(\cdot, \cdot)$ is a binary indicator function checking whether expert answers are equivalent to the original attribute value. An attribute pair is filtered as a hallucination 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 original attribute value. Table 1 shows raw data statistics. To illustrate the training set composition, we divided over 4,000 leaf categories into eight primary ones, selecting common attribute names for each and displaying them in Figure 3.

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

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 tree, sourced from AliExpress, with at most nine sibling categories sampled to form the choices. **(3) 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. **(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 15 instructions and 10 response templates were designed for each task to ensure diversity. Using a conversation format akin to Qwen-VL-Chat (Bai

	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']>, <['Bluetooth Version', '5.3']>, <['Wireless Type', 'Bluetooth']>], 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	Looking at the image along with the caption, what can you tell us about the product's attribute <Gender>? If it's indistinct, please reply with 'Unknown'.	The attribute value might be <WOMEN>.
AC	Does the attribute <Material> correctly bear the value <Silk>? In case of an inaccuracy, please advise on the accurate value.	No, the product's <Material> attribute is <Metal>.
CMC	From the selection below, pick the number corresponding to the category you deem most suitable for the product: 1: <Hair Ties>,2: <Headband>,3: <Hair Accessories Set>,4: <Hair Claw>,5: <Hair Scarf>.	It best fits into category <2: Headband>.

Table 2: Examples of each task in the training set, where the texts in blue are the given conditions and the texts in red are the ground truth answers. Here we omit the image input.

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.

et al., 2023), specific values are contained within <> to facilitate extraction in real scenarios. Table 2 offers several examples of each task, elucidating the details of these five tasks.

4 Benchmarking on Product Understanding Tasks

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., 2024), LLaVA-1.5(Liu et al., 2023), mPlug-Owl2(Ye et al., 2023), MiniGPT-4(Zhu et al., 2023), Qwen-VL-Chat(Bai et al., 2023) and GPT-4V(Achiam et al., 2023) to be the compared baselines. 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. For GPT-4V, we follow its default setting. The details can be seen in Table 7 in Appendix, and the prompts used for inference are shown in Table 8 in Appendix.

4.2 Datasets and metrics

PUMBENCH. We construct PUMBENCH to evaluate the capabilities of product understanding of

Tasks		InstBLIP	LLaVA	Mini	Owl2	Qwen-VL	GPT-4V	PumGPT
CG	Bleu_1	0.094	0.069	0.086	0.087	0.153	0.102	**0.383**
	ROUGE_L	0.120	0.073	0.080	0.092	0.148	0.110	**0.286**
	CIDEr	0.157	0.089	0.181	0.171	0.295	0.128	**0.987**
CC	Bleu_1	0.225	0.442	0.447	0.406	0.681	0.442	**0.934**
	ROUGE_L	0.383	0.370	0.578	0.388	0.687	0.337	**0.937**
	CIDEr	2.325	2.075	3.882	1.717	4.837	1.281	**8.595**
	Rec(%)	6.07	32.69	18.29	40.99	47.00	**92.09**	70.63
AI	Acc(%)	5.45	22.90	4.73	19.25	19.89	26.98	**60.70**
AC	F1(%)	66.77	59.25	42.39	58.12	77.79	71.38	**93.14**
	Prec(%)	50.43	54.77	65.39	60.09	69.20	81.11	**90.34**
	Rec (%)	98.77	64.53	31.37	56.29	88.81	63.74	**96.12**
	CAcc(%)	1.06	0.41	38.92	0.29	0.37	50.01	**60.52**
CMC	Acc(%)	24.82	32.55	39.45	61.73	46.39	82.55	**82.57**

Table 4: 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 Bleu_1, ROUGE_L and CIDEr.

PumGPT and the existing LVLMs. We collect 1.5k items and employ 2 PhD students to clean the hallucination attributes to construct the attribute inference test set according to their commonsense. We construct other task benchmarks as we did in building 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 the Mistral 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 Bleu_1(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 accurarcy(acc), F1, precision(prec), and recall(rec) to assess the attribution correction task and only accuracy on CMC task. All reported results are the averages of three separate runs.

5 Experimantal Results

5.1 Main Results on PumBench

Table 4 elucidates the comparative performance of PumGPT and other methodologies on PumBench. Overall, PumGPT demonstrates superior efficacy across a variety of tasks. Specifically, in the two caption-centric tasks, PumGPT excels in generating captions aligned with product attributes by distilling key characteristics from images. This proficiency translates into markedly higher scores on the ROUGE_L and CIDEr metrics, which evaluate recall and specific keyword utilization. In the caption completion 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 formulates the completed caption from most attribute values in the reference list rather than amending the original title, resulting in GPT-4V's underperformance in caption-related metrics.

Regarding the attribute-related tasks, PumGPT significantly surpasses both open-source models and GPT-4V. Notably, for attribute inference task, PUMGPT exceeds the performance of GPT-4V by a margin of over thirty percentage points, highlighting the difficulties that even advanced commercial models face in intricate product understanding tasks that require specialized domain knowledge. Furthermore, due to stringent compliance regulations, GPT-4V fails to address some test samples involving prohibited topics. In the attribute correction 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,

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
PUMGPT	**32.91**	35.49	**78.26**

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

thereby failing to furnish accurate values despite identifying erroneous attributes. Only MiniGPT-4 and GPT-4V can provide corrections, albeit still trailing PUMGPT.

In the product category multiple-choice question task, PUMGPT continued to demonstrate best-in-class performance. However, the margin was not as pronounced as in other tasks. GPT-4V's performance was comparable to PUMGPT, suggesting that this task, which fundamentally involves reasoning rather than domain-specific knowledge, presents a fairer comparative framework. This observation implies that GPT-4V's reasoning capabilities are superior. Despite training, our model only equaled GPT-4V's performance, indicating potential areas for further enhancement in this task.

5.2 Domain-level Results on Attribute Inference

We divided the attribute inference task test set into three major categories: Home, Electronics, and Clothing. Both the Home and Electronics domains encompass standardized goods. For these domains, most attributes and attribute values are predefined, allowing them to be directly extracted from product titles and specifications. Consequently, a product understanding model must have thoroughly internalized this information during training to accurately infer attribute values. In contrast, Clothing items represent non-standardized goods, characterized by attributes that may be custom-defined by vendors and subject to personal interpretation. For instance, the style of a garment could be described as both commute and casual. Therefore, product understanding models must learn the distribution of vendor-specific styles during training, suggesting a higher emphasis on fitting specific distributions.

Table 5 presents the performance outcomes of each method. Overall, PUMGPT consistently demonstrated superior performance. Within the

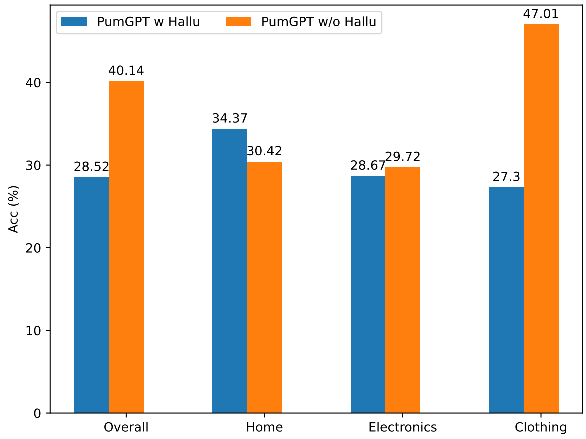


Figure 4: Ablation on hallucination filtering. Here we report the accuracy of the attribution inference task, where w Hallu means it was trained on the hallucination dataset and w/o Hallu means was trained on the hallucination-free dataset.

Home domain, our results exceeded those of GPT-4V by over three percentage points, and in the Electronics domain, the margin was 0.5 percentage points. PUMGPT outperformed the best Large Vision and Language Models (LVLMs) in standardized goods categories.

In the context of non-standardized goods, PUMGPT showcased exceptional performance on the attribute inference task by effectively learning from product data, thus capturing the distribution of vendor-desired descriptions. Conversely, models that lacked specific training only produced results reflecting their pre-training distributions. The performance of alternative models remains inadequate for application in real-world production environments.

5.3 Ablation on Hallucination Filtering

In Section 3.2, the crucial step involves filtering potentially hallucinatory attributes using our proposed multi-expert collaborative hallucination detection framework. For the task of attribute inference, PUMGPT achieved more than double the accuracy of GPT-4V. This significant performance improvement prompted an investigation to determine if it stemmed from our handling of hallucinations and to uncover the underlying causes.

We conducted an ablation experiment on hallucination processing. A subset of 600k entries was extracted from the original dataset of 663k entries. For the dataset containing hallucinations, up to eight attributes from each product's original attribute list were randomly sampled for training.

Models	F1	Prec	Rec	Acc
InstBLIP	0	0	0	89.53
LLaVA	17.67	20.95	15.27	88.30
Mini	0.75	4.44	0.41	90.10
Owl2	11.11	8.73	15.27	79.93
Qwen-VL	12.66	8.79	22.60	74.38
GPT-4V	29.69	19.33	**64.01**	74.47
PumGPT	**47.18**	**55.22**	41.12	**92.39**

Table 6: The evaluation on the rejection ability of all the compared methods.

For the hallucination-free dataset, the methods outlined in Section 3.2 were followed. The number of filtered attributes, including those designated as unknown, was strictly limited to eight. Both models underwent training for two epochs under identical training parameters.

As illustrated in Figure 4, PumGPT without hallucination data (w/o Hallu) showed significant performance improvement. The accuracy was classified into three primary categories, consistent with Section 5.2, to elucidate distinctions. In the standardized categories, performance differences between the models were marginal. In the Home category, PumGPT with hallucination data (w Hallu) outperformed PumGPT w/o Hallu by approximately four percentage points due to learning more attributes from the dataset. However, in the Clothing category, PumGPT w/o Hallu exceeded the other model by nearly 20 percentage points. The Clothing category predominantly includes non-standardized clothing items, with attributes often described subjectively. Consequently, PumGPT trained with hallucinated data may produce excessively imaginative yet inaccurate responses. In contrast, the model trained on the hallucination-free dataset can reduce such extrapolations, resulting in more accurate responses. Therefore, the processing of hallucinations is unequivocally vital for model training.

5.4 Evaluation on Rejection Ability

Large language models are acclaimed for their advanced text completion capabilities. However, they can sometimes produce incorrect information due to excessive associative reasoning. An effective model in practical applications should have the ability to refrain from responding when confronted with nonexistent or ambiguous attributes rather than providing a plausible but incorrect answer.

Consistent with our hallucination treatment within the training set, PumGPT defaults to the special attribute value "unknown" when queried about potentially hallucinatory attributes. As depicted in Table 6, accuracy (acc) is measured by labeling samples that refuse to respond as 1, and those that do not as 0. If no sample is refused, the acc would be 90%. Recall evaluates the recall rate among samples where a refusal is expected. Various models were assessed on their capacity to refuse to answer in attribute inference tasks. Open-source models like InsturctBLIP and MiniGPT-4 typically provide an actual value rather than refusing, inflating acc to around 90%. Therefore, examining F1, precision, and recall metrics is crucial as these indicate the susceptibility of these models to hallucinations, even when instructed to refuse.

In contrast, other open-source models attempt more refusals but achieve unsatisfactory accuracy. GPT-4V demonstrates higher refusal rates due to its conservative rules, but its overall accuracy is among the lowest. While our model's recall is lower than GPT-4V, it significantly excels in the overall F1 metric, demonstrating the effectiveness of our approach with "unknown" attributes in training sets. To enhance the model's refusal capabilities, employing preference learning algorithms such as PPO (Schulman et al., 2017) and DPO (Rafailov et al., 2023) may be necessary.

5.5 Case Study

We also perform a case study in Appendix A.3.

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 publishing processes, resulting in a dataset of approximately 663,000 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 general-purpose LVLMs, such as GPT-4V, across all tasks. 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.

References

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

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*.

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.

Danny Driess, F. Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Ho Vuong, Tianhe Yu, Wenlong Huang, Yevgen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Peter R. Florence. 2023. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning*.

William Fedus, Barret Zoph, and Noam M. Shazeer. 2021. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *J. Mach. Learn. Res.*, 23:120:1–120:39.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Qiang Liu, Kriti Aggarwal, Zewen Chi, Johan Bjorck, Vishrav Chaudhary, Subhojit Som, Xia Song, and Furu Wei. 2023. Language is not all you need: Aligning perception with language models. *ArXiv*, abs/2302.14045.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, L'elio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *ArXiv*, abs/2401.04088.

Trung-Hoang Le and Hady W Lauw. 2021. Explainable recommendation with comparative constraints on product aspects. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 967–975.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Annual Meeting of the Association for Computational Linguistics*.

Rongmei Lin, Xiang He, Jie Feng, Nasser Zalmout, Yan Liang, Li Xiong, and Xin Luna Dong. 2021. Pam: understanding product images in cross product category attribute extraction. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 3262–3270.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *ArXiv*, abs/1707.06347.

Yongliang Shen, Kaitao Song, Xu Tan, Dong Sheng Li, Weiming Lu, and Yue Ting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *ArXiv*, abs/2303.17580.

Keiji Shinzato, Naoki Yoshinaga, Yandi Xia, and Wei-Te Chen. 2022. Simple and effective knowledge-driven query expansion for qa-based product attribute extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 227–234.

Changfeng Sun, Han Liu, Meng Liu, Zhaochun Ren, Tian Gan, and Liqiang Nie. 2020. Lara: Attribute-to-feature adversarial learning for new-item recommendation. In *Proceedings of the 13th international conference on web search and data mining*, pages 582–590.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2014. Cider: Consensus-based image description evaluation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575.

Qifan Wang, Li Yang, Bhargav Kanagal, Sumit Sanghai, D Sivakumar, Bin Shu, Zac Yu, and Jon Elsas. 2020. Learning to extract attribute value from product via question answering: A multi-task approach. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 47–55.

Chenfei Wu, Sheng-Kai Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. 2023. Visual chatgpt: Talking, drawing and editing with visual foundation models. *ArXiv*, abs/2303.04671.

Huimin Xu, Wenting Wang, Xinnian Mao, Xinyu Jiang, and Man Lan. 2019. Scaling up open tagging from tens to thousands: Comprehension empowered attribute value extraction from product title. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5214–5223.

Jun Yan, Nasser Zalmout, Yan Liang, Christan Grant, Xiang Ren, and Xin Luna Dong. 2021. Adatag: Multi-attribute value extraction from product profiles with adaptive decoding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4694–4705.

Li Yang, Qifan Wang, Zac Yu, Anand Kulkarni, Sumit Sanghai, Bin Shu, Jon Elsas, and Bhargav Kanagal. 2022. Mave: A product dataset for multi-source attribute value extraction. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, pages 1256–1265.

Zhengyuan Yang, Linjie Li, Jianfeng Wang, Kevin Lin*, Ehsan Azarnasab, Faisal Ahmed, Zicheng Liu, Ce Liu, Michael Zeng, and Lijuan Wang. 2023. Mm-react: Prompting chatgpt for multimodal reasoning and action. *arXiv*.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. 2023. mplug-owl2: Revolutionizing multi-modal large language model with modality collaboration. *Preprint*, arXiv:2311.04257.

Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Jiao Qiao. 2023a. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *ArXiv*, abs/2303.16199.

Yupeng Zhang, Shensi Wang, Peiguang Li, Guanting Dong, Sirui Wang, Yunsen Xian, Zhoujun Li, and Hongzhi Zhang. 2023b. Pay attention to implicit attribute values: a multi-modal generative framework for ave task. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13139–13151.

Guineng Zheng, Subhabrata Mukherjee, Xin Luna Dong, and Feifei Li. 2018. Opentag: Open attribute value extraction from product profiles. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1049–1058.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Tiangang Zhu, Yue Wang, Haoran Li, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2020. Multimodal joint attribute prediction and value extraction for e-commerce product. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2129–2139.

Zihao Zhu, Mingda Zhang, Shaokui Wei, Bingzhe Wu, and Baoyuan Wu. 2024. Vdc: Versatile data cleanser based on visual-linguistic inconsistency by multimodal 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*.

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.

A.2 Model Details

The details of the model we compared and other generation configs are shown in Table 7.

Models	Size	LLM
InstBLIP	7B	Vicuna
LLaVA	7B	LLaMA
Mini	7B	LLaMA-2
Owl2	7B	LLaMA-2
Qwen-VL	7B	Qwen
GPT-4V	/	/
PumGPT	7B	Qwen

Table 7: The details of model size and their base LLMs.

A.3 Case Study

We also conducted a case study. Table 9 and Table 10 respectively display the results of all the models for a certain attribute on non-standardized and standardized products. It can be observed that most models are unable to infer results for the non-standardized product. These models either fail to generate the results or mistakenly output the entire product title while intending to 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 observed 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.

	Prompt
Question Gen	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. 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: <>. Attribute name: <>. Attribute value: <>. Question:
Answer Check	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). Attribute name: <>. Reference attribute value: <>. Candidate attribute value: <>. Judgement:
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: [<>]. The initial caption is: <>. Just answer the enhanced caption based on the initial caption with necessary attributes.
AI	The caption of the product in the image is . Please clarify the attribute of the product. Just respond with a simple phrase and respond unknown if you're not sure.
AC	Upon reviewing the product titled <> in the attached image, is the <> attribute correctly <>? Just respond yes or no. If this is incorrect, kindly provide the accurate value.
CMC	The caption of the product in the image is . Choose the most fitting category for the product: [<>]. Just answer the option number that you believe correct.

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

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".
Attribute name: Pattern	GPT-4V	Unknown.
	PUMGPT	The attribute value of the product may be <Print>.
	GT	Print.

Table 9: A case on a non-standardized product, where GT is the reference attribute value

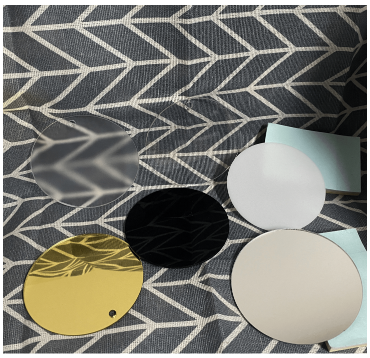
Product	Models	Answers
	InstructBLIP	No answer.
	LLaVA-1.5	Unknown.
	MiniGPT-4	Thank you for providing the image. The product in the image is an acrylic circle sheet disc.
	mPlug-Owl2	The model number of the product is not visible in the image.
	Qwen-VL-Chat	The model number of the product is not mentioned in the not provided.
	GPT-4V	Unknown.
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.	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>.
Attribute name: Model Number	**GT**	Acrylic Circle Sheet Disc Round Board.

Table 10: A case on a standardized product.