Adapting Large Vision-Language Models to Visually-Aware Conversational Recommendation

Hyunsik Jeon
University of California, San Diego
San Diego, CA, USA
hyjeon@ucsd.edu

Satoshi Koide
Toyota Motor Corporation
Nagakute, Japan
satoshi.koide@toyota.com

Yu Wang
University of California, San Diego
San Diego, CA, USA
yuw164@ucsd.edu

Zhankui He
Google DeepMind
Mountain View, CA, USA
zhh004@ucsd.edu

Julian McAuley
University of California, San Diego
San Diego, CA, USA
jmcauley@ucsd.edu

Abstract

Conversational recommender systems engage users in dialogues to refine their needs and provide more personalized suggestions. Although textual information suffices for many domains, visually driven categories such as fashion or home decor potentially require detailed visual information related to color, style, or design. To address this challenge, we propose LaViC (Large Vision-Language Conversational Recommendation Framework), a novel approach that integrates compact image representations into dialogue-based recommendation systems. LaViC leverages a large vision-language model in a two-stage process: (1) visual knowledge self-distillation, which condenses product images from thousands of tokens into a small set of visual tokens in a self-distillation manner, significantly reducing computational overhead, and (2) recommendation fine-tuning, which enables the model to incorporate both dialogue context and distilled visual tokens, providing a unified mechanism for capturing textual and visual features. To support rigorous evaluation of visually-aware conversational recommendation, we construct a new dataset by aligning *Reddit* conversations with *Amazon* product listings across multiple visually oriented categories (e.g., fashion, beauty, and home). This dataset covers realistic user queries and product appearances in domains where visual details are crucial. Extensive experiments demonstrate that LaViC significantly outperforms text-only conversational recommendation methods and open-source vision-language baselines. Moreover, LaViC achieves competitive or superior accuracy compared to prominent proprietary baselines (e.g., GPT-3.5-turbo, GPT-4o-mini, and GPT-4o), demonstrating the necessity of explicitly using visual data for capturing product attributes and showing the effectiveness of our vision-language integration. Our code and dataset are available at https://github.com/jeon185/LaViC.

CCS Concepts

• **Information systems → Recommender systems**.

KDD '25, August 3–7, 2025, Toronto, ON, Canada

ACM ISBN 979-8-4007-1454-2/2025/08
https://doi.org/10.1145/3711896.3736828

Keywords

Conversational Recommendation, Large Language Models, Large Vision-Language Models

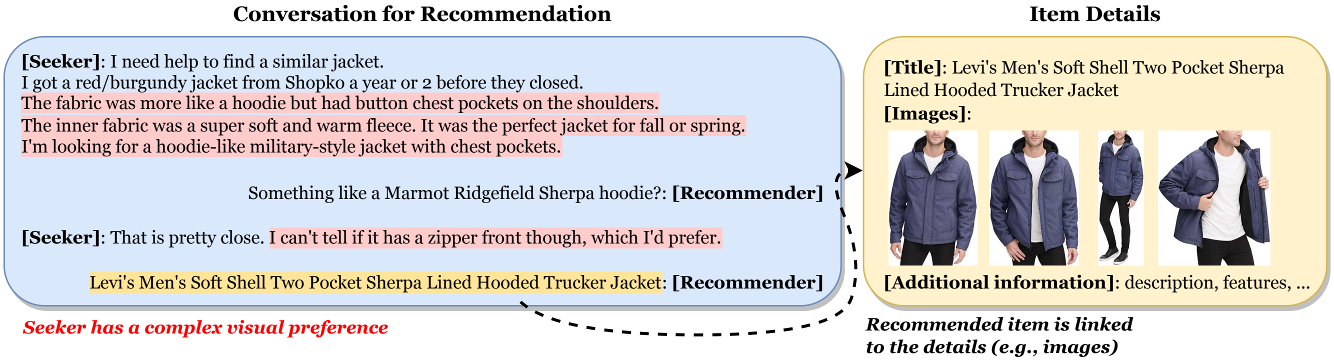
ACM Reference Format:
Hyunsik Jeon, Satoshi Koide, Yu Wang, Zhankui He, and Julian McAuley. 2025. Adapting Large Vision-Language Models to Visually-Aware Conversational Recommendation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3–7, 2025, Toronto, ON, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3711896.3736828

1 Introduction

Conversational recommendation has emerged as a promising framework for e-commerce and digital entertainment, offering an interactive channel for users to express their preferences through natural language [7, 17, 18, 29, 59]. Unlike traditional recommendation approaches that rely primarily on past interactions, conversational recommender systems aim to find relevant and personalized items by engaging in a dialogue with users.

Recently, *large language models* (LLMs) [6, 10, 17, 26, 50, 51] have demonstrated a strong ability to interpret user intentions within natural language conversational contexts. Their broad pre-training on extensive tasks allows them to handle complex linguistic features and incorporate specialized domain information when generating responses. This capability has led LLM-based approaches to outperform previous non-LLM methods in conversational recommendation [10, 17, 55], positioning LLMs as an essential component in the design of current conversational recommender systems.

Although text-based recommendation often suffices for many domains, certain product categories (e.g., fashion and home decor) rely heavily on visual features such as color, style, and overall design. Previous visually-aware recommender systems [16, 25, 44] have shown that images can substantially improve performance in visually oriented domains, indicating that purely textual descriptions may miss subtle aesthetic or functional details. For example, a user who requests a *"hoodie-like military-style jacket with chest pockets"* can narrow the options using textual information, yet verifying the exact silhouette or pocket arrangement depends on visual inspection (Figure 1). Thus, incorporating visual information helps resolve ambiguities and improves recommendation quality, making it more likely that a system identifies items aligned with the intended look or function of the user.


Conversation for Recommendation
Item Details
Seeker has a complex visual preference
Recommended item is linked to the details (e.g., images)

[Seeker]: I need help to find a similar jacket.
I got a red/burgundy jacket from Shopko a year or 2 before they closed.
The fabric was more like a hoodie but had button chest pockets on the shoulders.
The inner fabric was a super soft and warm fleece. It was the perfect jacket for fall or spring.
I'm looking for a hoodie-like military-style jacket with chest pockets.

Something like a Marmot Ridgefield Sherpa hoodie?: **[Recommender]**

[Seeker]: That is pretty close. I can't tell if it has a zipper front though, which I'd prefer.

Levi's Men's Soft Shell Two Pocket Sherpa Lined Hooded Trucker Jacket: **[Recommender]**

[Title]: Levi's Men's Soft Shell Two Pocket Sherpa Lined Hooded Trucker Jacket
[Images]:
[Additional information]: description, features, ...

Figure 1: The *Reddit-Amazon* dataset contains conversations between a seeker and a recommender. The seeker requests an item recommendation via text, focusing primarily on visual preferences. The item recommended to the seeker by the recommender is linked to detailed item information such as its title and images.

To fully integrate visual information into the conversation, one could resort to *large vision-language models* (VLMs) [34–36] that unify visual and textual inputs. However, such models generally tokenize each image into hundreds or thousands of tokens, incurring a steep computational load if multiple items are analyzed simultaneously. Moreover, naive end-to-end fine-tuning of massive multimodal architectures can lead to overfitting, particularly in specialized domains with limited labeled data. Finally, directly passing all retrieved items as raw image tokens can be prohibitively expensive in real-world deployments, indicating the need for a compact representation of visual content.

In this work, we address these challenges with a two-stage framework for *visually-aware conversational recommendation* built upon VLMs. First, we perform knowledge distillation to compress the high-dimensional encoding of each product image into a minimal set of visual tokens, preserving essential details of appearance while reducing computational overhead. Second, we adopt a recommendation fine-tuning procedure on a distilled vision-language model to jointly process textual queries and these compact visual tokens within a unified generative paradigm. This design balances computational feasibility with the need for detailed visual understanding. Furthermore, to enable rigorous empirical evaluation, we construct a new dataset, called *Reddit-Amazon* dataset, by aligning Reddit dialogues with Amazon product images, reflecting genuine user queries and realistic visual attributes seldom captured in existing benchmarks.

The main contributions of our work are as follows.

- We mitigate the token-explosion problem by condensing each product image into a minimal set of visual tokens, preventing the model from processing thousands of image tokens, and thus improving training stability, reducing overfitting, and ultimately enhancing recommendation accuracy.
- We propose a fine-tuning procedure on a distilled vision-language model, enabling it to jointly process textual dialogues and these compressed visual tokens within a unified generative framework for recommendation.
- We release *Reddit-Amazon*, a new dataset consisting of over 19K Reddit conversations (51K turns in total) aligned with Amazon product items. It spans three visually oriented categories (beauty, fashion, and home), linking each recommended item to its title and images. This dataset offers a richer testbed for visually-aware conversational recommendation.

2 Preliminaries

2.1 Problem Statement

In the *visually-aware conversational recommendation* task, we aim to recommend relevant items to a user (the seeker) through a multi-turn dialogue while considering both textual and visual features of the items. Let $\mathcal{I}$ be a set of items, where each item $i \in \mathcal{I}$ has a textual title $title_i$ and a single image $image_i$. A vocabulary $\mathcal{V}$ defines the tokens used for text. The conversation is $C = \{s_t\}_{t=1}^{T}$, where each utterance s_t is produced by the seeker or the recommender, drawn from $\mathcal{V}$. The seeker initially requests recommendations, and at each recommender turn k, the system produces a ranked list $\hat{\mathcal{I}}_k \subseteq \mathcal{I}$ that aligns with the true set $\mathcal{I}_k$. This set $\mathcal{I}_k$ embodies the user's current preferences, which may be represented through the conversation. In our setting, we assume $|\hat{\mathcal{I}}_k| = |\mathcal{I}_k| = 1$, which means there exists a single ground-truth item and we also aim to recommend a single item.

Generative models such as large language models (LLMs) or large vision–language models (VLMs) often produce unconstrained outputs, potentially recommending items that do not exist in $\mathcal{I}$. To avoid recommending non-existent items, we use a *candidate-based* approach: a retrieval module supplies a small set of candidate items, and the model selects the correct one. Furthermore, we prioritize *recommendation accuracy* over the generation of fully fluent dialogues, focusing on correct item selection rather than natural-sounding responses as in previous works [17, 18].

2.2 Large Vision-Language Models

We provide an overview of large vision-language models (VLMs) (e.g., LLaVA [34–36]), which serve as the backbone for our visually-aware conversational recommendation. These models combine a vision encoder (e.g., CLIP [46] or SigLIP [63]) with a large language

Table 1: Summary of *Reddit-Amazon* dataset. The *Reddit-Amazon* dataset consists of three sub-categories based on the type of recommended items: *beauty*, *fashion*, and *home*.

Dataset	# Conv.	# Turns	# Items	# Images
Beauty	7,672	22,966	5,433	28,082
Fashion	8,039	21,831	6,716	31,162
Home	3,701	6,675	3,077	18,505

model (LLM) (e.g., Vicuna [6] or Mistral [23]), enabling both image and text inputs to be handled in a unified transformer-based pipeline.

Vision encoder. A common approach for the vision encoder relies on Vision Transformers (ViT) [9], which divides an input image into R patches. Then, each patch is flattened and linearly embedded in a d dimensional vector. A special token (e.g., `[CLS]`) could be appended, giving $R+1$ tokens. These tokens are fed together into a transformer, yielding contextualized embeddings:

$$(\mathbf{p}_0, \ \ldots, \ \mathbf{p}_R) \ \mapsto \ (\mathbf{e}_0, \ \ldots, \ \mathbf{e}_R), \tag{1}$$

where $\mathbf{p}_0$ and $\mathbf{e}_0$ correspond to the `[CLS]` token and its embedding, respectively. Some frameworks (e.g., LLaVA-v1.5 [34] and LLaVA-v1.6 [35]) further subdivide the image into multiple sub-images, but the patch-oriented mechanism remains the same.

Aligning vision encoder with LLM. Once the vision encoder produces $\{\mathbf{e}_r\}_{r=0}^{R}$, a projector Ω_{proj} maps these embeddings into the LLM's space:

$$\mathbf{v}_r \ = \ \text{Proj}(\mathbf{e}_r; \, \Omega_{\text{proj}}), \tag{2}$$

where $\mathbf{v}_r \in \mathbb{R}^d$. Let Ω_{vision} encompass the vision encoder and projector. The text input $\mathcal{T}$ is tokenized into $\{\mathbf{x}_t\}_{t=1}^{|\mathcal{T}|}$. Then, LLM processes the following textual and visual embeddings:

$$[\mathbf{x}_1, \ldots, \mathbf{x}_{|\mathcal{T}|}, \, \mathbf{v}_0, \ldots, \mathbf{v}_R], \tag{3}$$

with Ω_{LM} denoting the LLM parameters. Over multiple layers of self-attention and feed-forward blocks [52], these embeddings merge into a contextualized representation.

Pretraining and fine-tuning. The pretraining of VLMs involves image-conditioned text generation [36]. At inference time, VLMs produce a token-level distribution:

$$P_{\Omega_{\text{LM}}+\Omega_{\text{vision}}}(\mathbf{y} \mid \mathcal{T}, \text{Image}), \tag{4}$$

where $\mathbf{y}$ are output tokens (e.g., a caption or an answer). Fine-tuning for downstream tasks (e.g., visual question answering or chatbot) updates either all parameters or a subset thereof via LoRA [21], preserving the model's pre-trained multimodal knowledge.

Challenges in using multiple images. Whereas standard VLM tasks (e.g., captioning) generally involve a single image, visually-aware conversational recommendation may require analyzing multiple items (e.g., ten or more) in one query. If each item image yields $R+1$ patch tokens (including a `[CLS]` token), ten items produce $10 \times (R+1)$ image tokens. Because self-attention scales quadratically with sequence length, this *token explosion* can exceed the LLM context window and memory budget. Moreover, it makes training unwieldy, causing the model to struggle with learning useful representations, and thereby dropping recommendation performance. For instance, LLaVA-v1.6 sets $R = 576$ for each of 5 sub-images and uses a 4,096 token context length, so handling ten images at once is infeasible under these constraints.

Table 2: Comparison of existing conversational recommendation datasets. Our *Reddit-Amazon* dataset consists of realistic conversations and focuses on visual-oriented domains such as beauty, fashion, and home. In the column of source, Syn., CS., and Nat. denote synthetic, crowd-sourced, and natural, respectively.

Dataset	#Conv.	#Turns	#Items	Domain	Source
FacebookRec [8]	1M	6M	-	Movies	Syn.
ReDial [29]	10K	182K	6.2K	Movies	CS.
GoRecDial [24]	9K	170K	-	Movies	CS.
OpenDialKG [45]	15K	91K	-	Movies, music, etc.	CS.
TG-ReDial [67]	10K	129K	-	Movies	Syn.
DuRecDial 2.0 [42]	16.5K	255K	-	Movies, music, etc.	CS.
CCPE-M [47]	502	11K	-	Movies	CS.
INSPIRED [14]	1K	35K	1.9K	Movies	CS.
Reddit-Movie$_{\text{base}}$ [17]	85K	133K	24.3K	Movies	Nat.
Reddit-Movie$_{\text{large}}$ [17]	634K	1.6M	51.2K	Movies	Nat.
U-NEED [41]	7K	53K	-	E-commerce	Nat.
E-ConvRec [22]	25K	775K	-	E-commerce	Nat.
HOOPS [12]	-	11.6M	-	E-commerce	Syn.
MGConvRec [60]	7K	73K	-	Restaurant	CS.
MMConv [32]	5K	39K	-	Travel	CS.
MobileConvRec [43]	12.2K	156K	1.7K	Music, sports, etc.	Syn.
Reddit-Amazon	19K	51K	15K	Beauty, fashion, home	Nat.

3 Reddit-Amazon Dataset

Although many conversational recommendation datasets have been introduced, most rely on crowd-sourced or synthetic dialogues (e.g., ReDial [29], GoRecDial [24], and TG-ReDial [67]). Crowd workers often lack specific personal interest, creating artificial queries and interactions [17], while fully synthetic dialogues are constructed from user behaviors or other data and may fail to capture the spontaneity and depth of real conversations. Datasets reflecting authentic user interactions (e.g., E-ConvRec [22] and Reddit-Movie [17]) tend to focus on domains such as movies or electronics, where visual attributes (e.g., color, style, or design) play a less prominent role. U-Need [41] covers categories like fashion and beauty, but does not include visual sources. These constraints of existing datasets limit their applicability to realistic visually-aware conversational recommendation.

To address these gaps, we collect real user data from Reddit via *pushshift.io*[1] in three visually oriented domains: beauty, fashion, and home. We first identify relevant posts and comments using GPT-3.5-turbo [50] and a filtering prompt (Table 9 in Appendix A), then link the mentioned items to the Amazon Reviews 2023 dataset [20]. The resulting *Reddit–Amazon* dataset is divided into three subsets (beauty, fashion, and home), each containing genuine user dialogues, ground-truth recommended items, and corresponding item features including images. Figure 1 provides an illustration and Table 1 presents key statistics for *Reddit–Amazon*, offering a

[1] https://pushshift.io

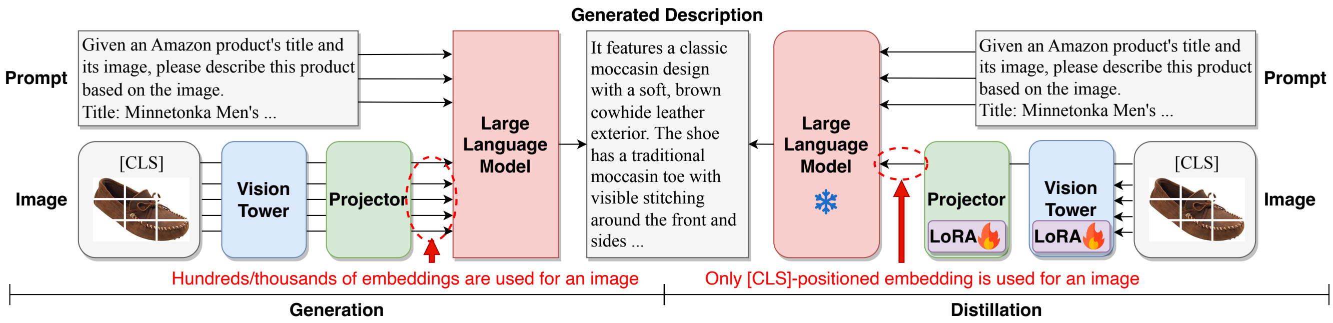

Generated Description
Prompt
Given an Amazon product's title and its image, please describe this product based on the image. Title: Minnetonka Men's ...
Image
[CLS]
Vision Tower
Projector
Large Language Model
It features a classic moccasin design with a soft, brown cowhide leather exterior. The shoe has a traditional moccasin toe with visible stitching around the front and sides ...
Large Language Model
Projector
LoRA
Vision Tower
LoRA
Given an Amazon product's title and its image, please describe this product based on the image. Title: Minnetonka Men's ...
Prompt
[CLS]
Image
Hundreds/thousands of embeddings are used for an image
Only [CLS]-positioned embedding is used for an image
Generation
Distillation

Figure 2: Illustration of *visual knowledge self-distillation.* Generation process (left): The vision tower and projector encode each sub-image into hundreds of patch embeddings (577 for each sub-image in LLaVA-v1.6), which are passed to the large language model (LLM) alongside a textual prompt. The LLM then produces a detailed product description focusing on visual features. Distillation process (right): We freeze the LLM and train only the vision tower and projector (via LoRA) to condense each sub-image into a single [CLS]-positioned embedding, yet still generate the same descriptive text. This reduces the token count from thousands to a handful, minimizing computational overhead while retaining essential visual information.

realistic benchmark for evaluating visually-aware conversational recommendation.

Table 2 compares *Reddit-Amazon* dataset with well-known conversational recommendation datasets. Early efforts (e.g., ReDial [29], GoRecDial [24], and TG-ReDial [67]) primarily focus on movies, often with crowd-sourced or synthetic dialogues. Recent datasets have expanded into broader domains (e.g., music, restaurants, and e-commerce) or used natural Reddit conversations [17, 22, 41], but still lack substantial visual information. In contrast, our *Reddit-Amazon* covers three visually oriented categories (beauty, fashion, and home), collecting 19K naturally occurring conversations with 51K turns. It aligns each conversation with 15K unique items and their images, capturing realistic discussions where users frequently refer to product appearances. This emphasis on visual details fills an important gap, allowing more comprehensive evaluations of visually-aware conversational recommender systems.

4 Visually-Aware Conversational Recommendation

In this section, we propose LaViC, a two-stage framework that addresses token explosion in *visually-aware conversational recommendation* by compressing images into fewer tokens and then fine-tuning a large vision-language model (VLM) for accurate recommendation.

4.1 Framework Overview

We build upon LLaVA-v1.6, which is recognized for its adaptability to a broad range of downstream tasks and demonstrates strong performance in multi-image scenarios [35]. LLaVA-v1.6 encodes each image by splitting it into 5 sub-images, each producing 577 tokens for a total of 2, 885 tokens per image. Analyzing multiple items in a single query can quickly surpass the 4,096 token context limit, making naive end-to-end training unstable and limiting recommendation accuracy (Section 2.2). To resolve these issues, LaViC employs two key stages:

- **Visual knowledge self-distillation (Section 4.2).** We freeze the parameters of the large language model and train only the vision module (vision tower and projector) to condense each item's 5 sub-images (5 × 577 tokens) into a small set of [CLS] embeddings. A task-oriented prompt, asking for visually relevant attributes, guides this distillation.
- **Recommendation fine-tuning (Section 4.3).** We then freeze the distilled vision module and fine-tune the large language model. Given the compressed image embeddings and textual context, the model predicts the correct item ID between 10 candidates, thus preventing the risk of generating nonexistent items.

4.2 Visual Knowledge Self-Distillation

We aim to produce the same image description with far fewer tokens. Figure 2 illustrates how we distill vision knowledge into fewer tokens in a self-distillation manner. Initially, the VLM sees each image with all sub-image tokens (i.e., 2, 885 tokens if 5 sub-images are each mapped to 577 tokens) and generates a detailed description. We then distill this capability so that the large language model (LLM) can regenerate the same description from a single [CLS] embedding per sub-image. This reduces the model's reliance on thousands of tokens, preventing overflow of the context window in candidate-based recommendations where multiple items appear in a single query.

Generation. In the generation process, the VLM (with parameters $\Omega_{\text{LM}} + \Omega_{\text{vision}}$) is given an instruction to generate the description of an item image, along with the entire set of sub-image tokens for $image_i$; the prompt used for generation is detailed in Table 10 (Appendix A). The model freely attends to all these tokens, eventually producing a textual description D_i (e.g., specifying color, material, or design). This step shows the ability of the VLM to generate visually rich output, but at the cost of processing thousands of tokens per image. We provide examples of these generated descriptions in Appendix B.

Distillation. Then, we freeze Ω_{LM} (the parameters of LLM) and keep only the vision-side parameters Ω_{vision} trainable to focus on distilling only visual capability. We utilize only the [CLS]-positioned embedding for each sub-image that must alone suffice to generate D_i. Formally, if $image_i$ is divided into 5 sub-images $\{I_{i,r}\}_{r=1}^{5}$, each sub-image $I_{i,r}$ yields $\mathtt{Tok}_{\text{vision}}(I_{i,r}) \in \mathbb{R}^{577\times d}$ in the original

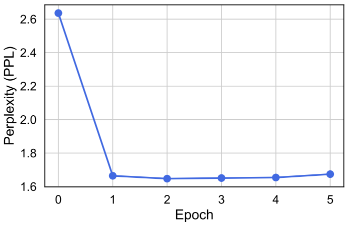


Figure 3: The validation perplexity reaches a plateau after 1–2 epochs.

model. Rather than passing all 577 tokens to the LLM, we extract the [CLS]-positioned embedding which we denote as $\mathbf{cls}_{i,r} \in \mathbb{R}^d$. Concatenating $\{\mathbf{cls}_{i,r}\}$ across all 5 sub-images yields 5 embeddings per entire image. We replace the original sub-image tokens in the input sequence with these 5 [CLS]-positioned embeddings and then prompt the frozen LLM to generate D_i.

Training objective. We train the vision module so that the LLM's output, given only 5 [CLS]-positioned embeddings per image, reproduces the same description D_i that was previously generated from all sub-image tokens. This drives each $\mathbf{cls}_{i,r}$ to encapsulate the most crucial visual details. Concretely, we optimize:

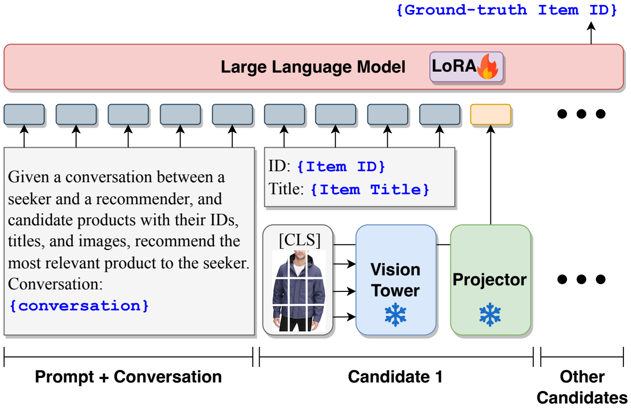
$$\min_{\Omega_{\text{vision}}} \sum_i -\log P_{\Omega_{\text{LM}}+\Omega_{\text{vision}}}\left(D_i \mid \mathcal{T}_{\text{desc}}, \{\mathbf{cls}_{i,r}\}_{r=1}^{5}\right), \tag{5}$$

where $\mathcal{T}_{\text{desc}}$ is the textual prompt asking for visually relevant attributes. We adopt LoRA [21] in the vision tower and the projector to keep the overhead parameter minimal. Figure 3 shows the perplexity (PPL) on a held-out set of 512 images and their generated descriptions; it typically converges after 1–2 epochs, showing the effectiveness of our focus on a small set of trainable parameters. We select the checkpoint with the lowest PPL (2 in our case) for the subsequent recommendation tasks.

After this self-distillation, each image is effectively represented by only 5 [CLS]-positioned embeddings rather than thousands of tokens, yet the frozen LLM can still reconstruct D_i. This preserves descriptive power with minimal visual tokens, laying the groundwork for candidate-based conversational recommendation, where multiple images may appear in a single query, without exceeding the model context.

4.3 Recommendation Fine-Tuning

Once the vision parameters Ω_{vision} are obtained after the visual knowledge self-distillation (Section 4.2), we focus on generating accurate recommendations for a given prompt. In turn k of a conversation, the model receives a dialogue context $\mathbf{C} = \{s_t\}_{t=1}^{k-1}$ between the seeker and the recommender. A retrieval module (e.g., SBERT [48]) then presents 10 candidate items $\{i_1, \ldots, i_{10}\}$. Each item i_j is represented by a textual title $title_{i_j}$ and 5 distilled [CLS]-positioned embeddings $\{\mathbf{cls}_{i_j,r}\}_{r=1}^{5}$. We make sure that exactly one candidate i^* is correct for each training example. In other words, if there are multiple ground-truths, we split the example into multiple training instances. Besides, if i^* is initially absent in the candidates, we swap out one negative candidate for i^* to ensure precisely one ground-truth item. We also randomly shuffle the candidate items to avoid any bias in their order.



Given a conversation between a seeker and a recommender, and candidate products with their IDs, titles, and images, recommend the most relevant product to the seeker.
Conversation:
{conversation}

Figure 4: Illustration of recommendation fine-tuning. We integrate the compressed embeddings with conversation and item IDs/titles for recommendation. We then train the large language model (LLM) using LoRA while fixing the parameters of vision tower and projector.

Prompt setting. Figure 4 illustrates how we feed textual information and each candidate's compressed image embeddings into the LLM. For each item i_j, we concatenate its ID id_{i_j} and textual title $title_{i_j}$ with its 5 [CLS]-positioned embeddings $\{\mathbf{cls}_{i_j,r}\}_{r=1}^{5}$ as follows:

$$\mathbf{X}_{i_j} = \text{Concat}\Big(id_{i_j}, title_{i_j}, \ \mathbf{cls}_{i_j,1}, \ldots, \mathbf{cls}_{i_j,5}\Big). \tag{6}$$

We then supply $\{\mathbf{X}_{i_j}\}_{j=1}^{10}$ and the tokenized conversation context with a task description prompt $\mathcal{T}_{\text{conv}}$ to the LLM, which is instructed to output a short item ID id_{i_j} (e.g., 10 characters) for i^*. This ID-based approach normalizes the system's output, avoiding hallucinated item names that do not appear in the entire set of items $\mathcal{I}$. The prompt used for recommendation is detailed in Table 11 (Appendix A).

Training objective. We train the LLM so that it generates the exact ID of the ground-truth item assuming that the vision module is already trained to extract crucial visual features. Each instance contains exactly one ground-truth item (i^*) and nine negatives, restricting the output to a concise ID. Let id_{i^*} denote the ID of the ground-truth item. We aim to generate id_{i^*} from the dialogue context $\mathcal{T}_{\text{conv}}$ and the representation of each candidate $\{\mathbf{X}_{i_j}\}_{j=1}^{10}$. Specifically, we optimize:

$$\min_{\Omega_{\text{LM}}} \sum_{(\mathcal{T}_{\text{conv}}, \mathcal{I}_{\text{cand}})} -\log P_{\Omega_{\text{LM}}+\Omega_{\text{vision}}}\left(ID_{i^*} \,\Big|\, \mathcal{T}_{\text{conv}}, \{\mathbf{X}_{i_j}\}_{j=1}^{10}\right), \tag{7}$$

where $\mathcal{I}_{\text{cand}}$ denotes the candidate items in an example. We adapt LoRA exclusively to the LLM's parameters Ω_{LM} while fixing the distilled vision module Ω_{vision}.

In inference, the core visual features of each candidate item are compactly encoded as 5 [CLS]-positioned embeddings, while the LLM receives the dialogue context, item IDs, and titles. Drawing on these visual and textual inputs, the model determines which candidate is best aligned with the user's preferences and outputs the corresponding item ID. Although the vision module supplies only a handful of tokens per image, it preserves sufficient detail

Table 3: Performance comparison of LaViC with open-source baselines. Gray colored methods require a candidate retrieval to generate answers. We use SBERT as the common retrieval since it shows the best performance among the open-source retrievals. Bold and underline indicate the best and the second-best, respectively.

Method	Beauty		Fashion		Home	
	HR@1	VR	HR@1	VR	HR@1	VR
Retrieval Baselines (item title)						
BM25	0.0169	-	0.0140	-	0.0479	-
SBERT	0.0551	-	0.0681	-	0.2166	-
RoBERTa$_{\text{large}}$	0.0640	-	0.0631	-	0.1814	-
SimCSE$_{\text{large}}$	0.0326	-	0.0301	-	0.0957	-
BLaIR$_{\text{base}}$	0.0371	-	0.0441	-	0.1335	-
Generative Baselines (item title) + SBERT						
Vicuna-v1.5	0.0533	0.9870	0.0481	0.9903	0.1184	1.0000
LLaVA-v1.5	0.0476	0.9896	0.0441	0.9855	0.0932	1.0000
LLaVA-v1.6	0.0770	0.9870	0.0827	0.9867	0.2030	0.9919
Generative Baselines (item title and image) + SBERT						
LLaVA-v1.5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
LLaVA-v1.6	0.0584	0.9741	0.0459	0.9843	0.1089	0.9919
Proposed Method (item title and image) + SBERT						
LaViC (ours)	**0.1187**	0.9702	**0.1232**	0.9298	**0.3197**	0.9892
Improvement	**+54.2%**	-	**+49.0%**	-	**+47.6%**	-

Table 4: Performance comparison of LaViC with proprietary baselines. We use SBERT as a candidate retrieval.

Method	Beauty		Fashion		Home	
	HR@1	VR	HR@1	VR	HR@1	VR
Generative Baselines (item title) + SBERT						
GPT-3.5-turbo	0.0968	0.9935	0.0977	0.9903	0.2343	1.0000
GPT-4o-mini	0.1213	1.0000	0.1160	0.9927	0.3258	0.9973
GPT-4o	0.1271	0.9987	0.1278	0.9976	0.3350	1.0000
Generative Baselines (item title and image) + SBERT						
GPT-4o-mini	0.1081	0.9974	0.1098	0.9927	0.2861	0.9946
GPT-4o	0.1160	0.9974	0.1231	0.9939	0.3308	0.9973
Proposed Method (item title and image) + SBERT						
LaViC (ours)	0.1187	0.9702	0.1232	0.9298	0.3197	0.9892

for domains with high visual complexity as shown in Figure 3. Simultaneously, the LLM remains within its context limits and can robustly integrate multi-item information. Thus, LaViC achieves accurate recommendations without incurring token overflow, making it suitable for practical candidate-based pipelines.

5 Experiments

5.1 Experimental Setup

Datasets. We use the newly constructed *Reddit-Amazon* datasets, which consist of three different domains: beauty, fashion, and home. Refer to Section 3 for details and Table 1 for a summary. For each domain, we divide conversations into training, validation, and test sets in an 8:1:1 ratio. For multiple ground-truth items, we construct the conversation as multiple examples to make each example have a single ground-truth.

Baselines. We compare our proposed method with two categories of baselines: retrieval-based and generative methods. Although there exist earlier knowledge graph-based conversational recommendation methods (e.g., KBRD [4], KGSF [66], and UniCRS [56]), they rely heavily on knowledge graphs and have mainly been evaluated on movie domains. Generalizing such approaches to diverse product categories (beauty, fashion, home) is non-trivial. Moreover, prior studies indicate that zero-shot large language models (LLMs) often surpass such knowledge graph-based conversational recommender systems on Reddit dialogues [17]. Thus, we focus on retrieval-based and generative baselines that can flexibly handle our *Reddit-Amazon* dataset.

(1) Retrieval methods. All following methods except BM25 leverage pre-trained models to encode the conversation text and item titles, then rank items by cosine similarity. The top-ranked item is returned as the recommendation.

- **BM25** [49]: It scores titles by token overlap with the conversation text.
- **SBERT** [48]: It generates sentence embeddings via a siamese BERT-based approach.
- **RoBERTa$_{\text{large}}$** [39]: It replicates BERT with careful hyperparameter tuning and larger training data.
- **SimCSE$_{\text{large}}$** [13]: It employs a contrastive learning objective for sentence embeddings.
- **BLaIR$_{\text{base}}$** [20]: It is a specialized sentence embedding model for recommendation, trained on a large-scale Amazon review dataset to capture item-text correlations.
- **OpenAI-emb$_{\text{large}}$**[2]: It is a proprietary OpenAI[3] model, one of the most powerful encoders for many complex textual contexts.

(2) Generative methods. Each method first retrieves the top-10 candidates (using one of the above retrieval models), then employs a shared prompt template to generate a single recommendation from those candidates.

- **Vicuna-v1.5** [6]: It is a fine-tuned version of LLaMA [36] on user-shared conversations from ShareGPT.
- **LLaVA-v1.5** [34]: It extends Vicuna for vision-language tasks.
- **LLaVA-v1.6** [35]: It is fine-tuned for multi-image tasks with using Mistral [23] as an LLM.
- **GPT-3.5-turbo** [50], **GPT-4o-mini**, and **GPT-4o**: These are OpenAI proprietary models with powerful abilities to solve complex tasks in a zero-shot environment [2].

Implementation details. We conduct all experiments on a single *NVIDIA A100 40GB* GPU. We obtain open-source pretrained models (SBERT, RoBERTa, SimCSE, BLaIR, Vicuna, and LLaVA) from the official HuggingFace[4] repositories, while GPT embeddings, GPT-3.5-turbo, GPT-4o-mini, and GPT-4o are accessed via the OpenAI API. All open-source generative methods use the 7B parameter scale (i.e., Vicuna-v1.5-7B, LLaVA-v1.5-7B, LLaVA-v1.6-7B, and LaViC-7B (ours)) to enable both training and evaluation on a single GPU. For

[2] https://platform.openai.com/docs/guides/embeddings
[3] https://openai.com
[4] https://huggingface.co

Table 5: Performance comparison of LaViC with open-source baselines. Gray colored methods require a candidate retrieval to generate answers. We use OpenAI-emb$_{\text{large}}$, which is one of the most powerful proprietary encoder, as the common retrieval. Bold and underline indicate the best and the second-best, respectively.

Method	*Beauty*		*Fashion*		*Home*	
	HR@1	**VR**	**HR@1**	**VR**	**HR@1**	**VR**
Retrieval Baselines (item title)						
BM25	0.0169	-	0.0140	-	0.0479	-
SBERT	0.0551	-	0.0681	-	0.2166	-
RoBERTa$_{\text{large}}$	0.0640	-	0.0631	-	0.1814	-
SimCSE$_{\text{large}}$	0.0326	-	0.0301	-	0.0957	-
BLaIR$_{\text{base}}$	0.0371	-	0.0441	-	0.1335	-
OpenAI-emb$_{\text{large}}$	<u>0.1461</u>	-	<u>0.1393</u>	-	<u>0.3224</u>	-
Generative Baselines (item title) + OpenAI-emb$_{\text{large}}$						
Vicuna-v1.5	0.0728	0.9819	0.0718	0.9867	0.1360	1.0000
LLaVA-v1.5	0.0762	0.9832	0.0768	0.9903	0.1114	0.9946
LLaVA-v1.6	0.0972	0.9870	0.1150	0.9867	0.2430	0.9839
Generative Baselines (item title and image) + OpenAI-emb$_{\text{large}}$						
LLaVA-v1.5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
LLaVA-v1.6	0.0641	0.9728	0.0857	0.9806	0.1646	0.9946
Proposed Method + OpenAI-emb$_{\text{large}}$						
LaViC (ours)	**0.1743**	0.9676	**0.1787**	0.9455	**0.3537**	0.9892
Improvement	**+19.3%**	-	**+28.3%**	-	**+9.7%**	-

generative approaches (including LaViC), we adopt a candidate-based pipeline: (1) retrieve the top-10 items for each conversation, then (2) generate a single recommendation via an LLM prompt. We specifically employ SBERT and OpenAI-emb$_{\text{large}}$ as retrieval methods: SBERT gives the best performance among open-source retrieval baselines, whereas OpenAI-emb$_{\text{large}}$ performs best when proprietary methods are considered.

Hyperparameters and training details. In the visual knowledge self-distillation (Section 4.2), we set the maximum output length to 128 tokens during the generation process. For the distillation process, we use a batch size of 4 and apply LoRA [21] to the vision tower and projector with $r = 8$, $\alpha = 32$, and a 0.1 dropout rate. We explore learning rates in $\{5\times10^{-5}, 10^{-5}, 5\times10^{-6}, 10^{-6}\}$ and weight decays in $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 0\}$, training for up to 5 epochs and picking the best checkpoint based on validation performance. In practice, results typically peak around epoch 2.

In the recommendation fine-tuning (Section 4.3), we limit the input length to 2K tokens, set the batch size to 1, and again apply LoRA with $r = 8$, $\alpha = 32$, and 0.1 dropout rate. We repeat the same grid search for learning rates and weight decays, training for up to 5 epochs and selecting the best model by validation performance. Most models converge by epoch 1 or 2 under these conditions.

Evaluation Metrics. Our primary metric is *HitRatio@1 (HR@1)*, which measures the performance of the recommendation by computing the proportion of conversations where the prediction is equal to the ground-truth item. For generative methods, we additionally report a *ValidRatio (VR)* following a previous work [31], defined as the fraction of generated responses that match one of the candidate items; it measures the model's adherence to the prompt instructions.

Table 6: Performance comparison of LaViC with proprietary baselines. We use OpenAI-emb$_{\text{large}}$ as a candidate retrieval.

Method	*Beauty*		*Fashion*		*Home*	
	HR@1	**VR**	**HR@1**	**VR**	**HR@1**	**VR**
Generative Baselines (item title) + OpenAI-emb$_{\text{large}}$						
GPT-3.5-turbo	0.1449	0.9987	0.1523	0.9964	0.2997	1.0000
GPT-4o-mini	0.1809	1.0000	0.1755	0.9976	0.3552	1.0000
GPT-4o	0.2005	0.9948	0.1944	0.9939	0.4055	1.0000
Generative Baselines (item title and image) + OpenAI-emb$_{\text{large}}$						
GPT-4o-mini	0.1667	0.9974	0.1623	0.9976	0.3526	1.0000
GPT-4o	0.1914	0.9974	0.1942	0.9939	0.3980	1.0000
Proposed Method + OpenAI-emb$_{\text{large}}$						
LaViC (ours)	0.1743	0.9676	0.1787	0.9455	0.3537	0.9892

5.2 Overall Performance

SBERT as a retrieval. We compare our method with two different categories of baselines: open-source baselines and proprietary baselines. For generative baselines and our method, we rely on a candidate-based recommendation setup with 10 items, using SBERT [48] which shows the highest performance between retrievals. Generative baselines are tested under two configurations: using only item titles (*item title*) or using both titles and images (*item title and image*).

Table 3 shows the comparison of LaViC with open-source baselines. *Retrieval-based* baselines (BM25, SBERT, RoBERTa$_{\text{large}}$, SimCSE$_{\text{large}}$, and BLaIR$_{\text{base}}$) rank items by textual similarity only. Their effectiveness varies by domain. For example, in Home, SBERT achieves $HR@1 = 0.2166$, which exceeds LLaVA-v1.6 ($HR@1 = 0.2030$) when both rely on titles alone. In contrast, in Beauty and Fashion, LLaVA-v1.6 with titles outperforms these retrieval methods. We also observe that incorporating images without specialized handling can degrade the performance of generative models, which is consistent with a previous work [40]. LLaVA-v1.6 shows reduced accuracy in most settings when images are used, and LLaVA-v1.5 fails in multi-image candidate scenarios dropping VR to zeros on all domains; LLaVA-v1.6 is trained under multi-image tasks, while LLaVA-v1.5 is not. In contrast, LaViC achieves the highest HR@1 in all domains, up to 54.2% higher performance than the second-best. Furthermore, LaViC maintains a high VR, indicating consistency in generating valid outputs. These results suggest that compressing each item image into several `[CLS]`-positioned embeddings effectively preserves critical visual details while preventing token explosion, thus yielding greater recommendation accuracy in visually intensive domains.

Table 4 compares LaViC with proprietary baselines. They also rely on SBERT for candidate retrieval and then generate an item ID using textual titles or both titles and images. GPT-4o generally achieves the highest HR@1 in all domains, showing its powerful zero-shot performance in complex downstream tasks. GPT-4o-mini and GPT-4o both lose accuracy when images are included, indicating that multi-image processing can be challenging in complex conversational recommendations. Our method obtains performance

Table 7: Ablation study. SBERT is used for retrieval. *o.o.m.* indicates out-of-memory errors that prevent running under our single-GPU setup.

Method	*Beauty*		*Fashion*		*Home*	
	HR@1	VR	HR@1	VR	HR@1	VR
Entire tokens (5 × 577)	0.0256	0.9456	*o.o.m.*		*o.o.m.*	
w/o images	0.0972	0.9767	0.1022	0.9358	0.2944	0.9946
w/o self-distillation	0.0842	0.9793	0.1084	0.9649	0.2861	0.9973
LaViC (ours)	**0.1187**	0.9702	**0.1232**	0.9298	**0.3197**	0.9892

on par with GPT-4o-mini or GPT-4o, and exceeds GPT-3.5-turbo in all domains, despite relying on 7B-parameters. These results show that distilled vision embeddings coupled with an LLM can handle multi-item visual contexts as effectively as or better than larger commercial systems while mitigating token explosion. As a result, LaViC provides a strong balance of visual representation efficiency and recommendation accuracy in domains where product appearance strongly influences user choice.

OpenAI-emb$_{\text{large}}$ as a retrieval. We also compare performance using OpenAI-emb$_{\text{large}}$, which is known to deliver strong performance in various tasks, as a retrieval. Tables 5 and 6 report experimental results when using OpenAI-emb$_{\text{large}}$ as the retrieval instead of SBERT. We compare LaViC against both open-source and proprietary baselines in text-only (*item title*) and combining text and image (*item title and image*) settings.

Table 5 compares LaViC with open-source models. OpenAI-emb$_{\text{large}}$ greatly improves initial retrieval quality when it is compared to SBERT (Table 3). In particular, even the generative baselines underperform this retrieval model. Nevertheless, LaViC maintains a performance advantage across all domains (beauty, fashion, and home), outperforming the strongest open-source competitor by up to 28.3% in the fashion domain. When coupled with images, other methods often struggle to effectively process multiple images and exhibit lower accuracy.

Table 6 compares LaViC with GPT-3.5-turbo, GPT-4o-mini, and GPT-4o under using OpenAI-emb$_{\text{large}}$ as the retrieval. GPT-4o still achieves the highest overall HR@1. However, LaViC remains competitive in all domains and surpasses GPT-3.5-turbo. These results show that the design of LaViC to address token explosion preserves strong recommendation accuracy, even when a more powerful retrieval approach is used.

5.3 Ablation Study

Table 7 compares LaViC with three variants, each omitting a main component of our framework. *Entire tokens* (5 × 577) applies recommendation fine-tuning to LLaVA-v1.6 directly, processing all 2, 885 tokens per image. In the beauty domain, this configuration requires about one week for a single epoch on two A100 GPUs and yields low accuracy (HR@1=0.0256). Given these resource constraints, we cannot run the fashion and home configurations on our single-GPU setup, so we mark them as *o.o.m.*. In *w/o images*, the model eliminates visual input, relying solely on item titles, and achieved higher accuracy than *Entire tokens* (e.g., HR@1 = 0.0972 in beauty). However, this shows a lower HR@1 than LaViC in all domains, since it ignores visual characteristics. The *w/o self-distillation* preserves [CLS] extraction per sub-image but bypasses our visual knowledge distillation stage, further improving over *Entire tokens* but still underperforming LaViC. In contrast, LaViC incorporates both visual knowledge self-distillation and recommendation fine-tuning, reducing each image to 5 [CLS]-positioned embeddings with minimal loss of visual detail. LaViC attains the best HR@1 and strong VR across all domains, demonstrating token compression via self-distillation is crucial for balancing memory usage, training time, and accuracy in visually-aware conversational recommendations.

Table 8: Separate dataset vs. combined dataset.

Method	*Beauty*		*Fashion*		*Home*	
	HR@1	VR	HR@1	VR	HR@1	VR
LaViC-*separate*	**0.1187**	0.9702	**0.1232**	0.9298	**0.3197**	0.9892
LaViC-*combined*	0.1021	0.9585	0.1220	0.9673	0.3141	0.9651

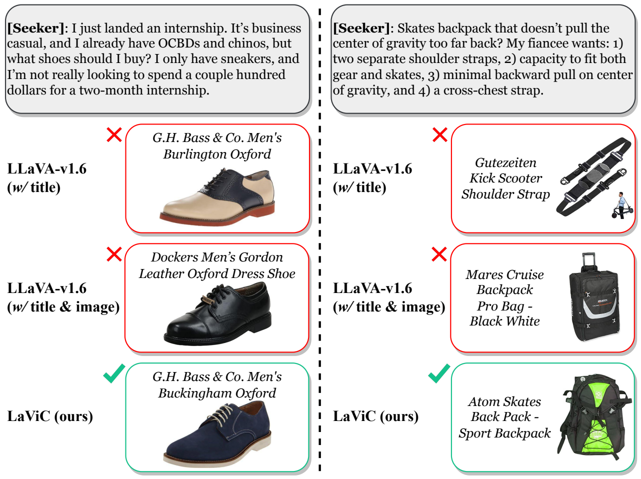
5.4 Further Analysis

5.4.1 Separate dataset vs. combined dataset. Table 8 compares two training strategies for LaViC. In LaViC-*separate* training, we learn a different model in each domain (beauty, fashion, and home) and then evaluate it in the same domain. In LaViC-*combined* training, we merge all three domains' data into a single training set and test on each domain separately. The results show that separate training yields slightly higher HR@1 than the combined approach. We hypothesize that combining data from multiple domains does not confer immediate benefits because the recommended items do not overlap between beauty, fashion, and home, and each domain's user preference is distinct. These factors limit cross-domain information sharing. Another consideration is that we fixed a single set of hyperparameters for the merged dataset, which may not optimally fit the specialized characteristics of each domain. However, our dataset remains small in size compared to many industrial-scale corpora. It is plausible that with more extensive data and broader conversational diversity, the model might learn cross-domain representations that improve performance. Hence, future studies could explore whether larger-scale domain mixtures further enhance performance in visually-aware conversational recommendations.

5.4.2 Case study. Figure 5 illustrates two cases where LaViC outperforms two variants of LLaVA-v1.6: one using only titles (*w/* title) and the other combining titles and images (*w/* title & image). We use OpenAI-emb$_{\text{large}}$ as the retriever to select 10 relevant candidates for each scenario, showing how well each method handles subtle differences among highly relevant items. See Table 5 for a quantitative comparison.

(a) Same brand but different style. The seeker wants a shoe for business-casual wear, and the ground-truth item is from "G.H. Bass & Co. Men's Buckingham Oxford" with a casual design. LLaVA-v1.6 (*w/* title) recommends a product of the same brand, but with a more formal look. LLaVA-v1.6 (*w/* title & image) fails to match the brand and casual style the seeker prefers. In contrast, LaViC selects an item from "G.H. Bass & Co." that better matches the intended style, relying on subtle visual features missing from the textual title.

(b) Specific requirements. The seeker needs a backpack with two separate shoulder straps, a cross-chest strap, and enough space for



[Seeker]: I just landed an internship. It's business casual, and I already have OCBDs and chinos, but what shoes should I buy? I only have sneakers, and I'm not really looking to spend a couple hundred dollars for a two-month internship.

[Seeker]: Skates backpack that doesn't pull the center of gravity too far back? My fiancee wants: 1) two separate shoulder straps, 2) capacity to fit both gear and skates, 3) minimal backward pull on center of gravity, and 4) a cross-chest strap.

LLaVA-v1.6 (w/ title)
G.H. Bass & Co. Men's Burlington Oxford
LLaVA-v1.6 (w/ title & image)
Dockers Men's Gordon Leather Oxford Dress Shoe
LaViC (ours)
G.H. Bass & Co. Men's Buckingham Oxford
LLaVA-v1.6 (w/ title)
Gutezeiten Kick Scooter Shoulder Strap
LLaVA-v1.6 (w/ title & image)
Mares Cruise Backpack Pro Bag - Black White
LaViC (ours)
Atom Skates Back Pack - Sport Backpack

(a) Same brand but different style (b) Specific requirements

Figure 5: Two cases comparing LaViC with LLaVA-v1.6. Both cases are selected from *fashion* domain. The ✓ indicates a correct recommendation, while the ✗ denotes an incorrect one. (a) LaViC identifies a more casual "G.H. Bass & Co. Men's Buckingham Oxford", whereas LLaVA-v1.6 either suggests a more formal shoe under the same brand or a different brand entirely. (b) LaViC recommends a backpack that meets the user's specific strap requirements (two separate shoulder straps and a cross-chest strap), which LLaVA-v1.6 (*w/* title & image) fails to satisfy, even with image input.

skates. LLaVA-v1.6 (*w/* title) recommends a strap-only accessory instead of the full backpack the seeker requested. LLaVA-v1.6 (*w/* title & image) does recommend a backpack, but lacks the required strap configuration. In contrast, LaViC identifies a backpack with two distinct shoulder straps and a cross-chest strap, leveraging the visual information in the images.

6 Related Work

Conversational recommendation. Early conversational recommender systems (CRS) rely on handcrafted dialogue flows and critiquing mechanisms. Users iteratively refine recommendations by providing feedback on item attributes (e.g., *"show me something with a lower price and longer battery life"*). These methods are largely template or rule-based [3, 7, 19, 27, 28, 30, 58, 65], requiring predefined responses for each user critique, and thus limiting flexibility. With the advent of neural approaches, CRS begins to leverage natural language understanding and generation. Recent models integrate techniques like knowledge graphs, reinforcement learning, and memory networks to conduct multi-turn dialogues that both converse and recommend items. Such systems generate more fluent, context-aware responses than rigid templates. For example, KBRD [4], KGSF [66], and UniCRS [56] incorporate external knowledge to enrich the dialogue, and TSCR [68] uses transformer-based architectures for better context understanding. Recently, large language models (LLMs) have been actively applied to CRS [11, 17, 55]. In particular, previous analyses show that LLMs can outperform specialized dialogue recommender systems even without fine-tuning [17]. These analyses show the promise of using LLMs as the backbone of CRS. However, all of these efforts predominantly handle textual interactions. Our approach is among the first to incorporate image content directly into CRS, enabling recommendations based on visual preferences. In contrast to previous CRS that do not consider images, we integrate visual information directly into the model, allowing us to understand and respond to the visual preferences of users.

Visually-aware recommendation. Visually-aware recommender systems (VARS) incorporate item images to capture style, design, or aesthetic aspects relevant to user preferences [37]. Early work integrated features extracted by pre-trained convolutional neural networks (CNNs) into collaborative filtering, improving recommendations in domains such as fashion, where visual features play a critical role [16, 25, 38, 44, 54]. Later studies developed end-to-end pipelines that jointly optimized user-item interactions and image representation learning [15, 57]. In parallel, various architectures have been explored, including graph neural networks [53, 57] and attention-based models [5, 33, 64]. Recently, large vision-language models (VLMs) such as CLIP [46] and VLMo [1] have extended these efforts by mapping images and text into a shared embedding space [40, 62]. For example, Rec-GPT4V [40] leverages GPT-4Vision [61] for zero-shot item ranking by prompting the model with each product's image. However, this approach can become inefficient for multi-item scenarios due to the high token overhead. In contrast, our method encodes each product image into a small set of embeddings, retaining essential visual information without incurring excessive token costs. Our design is well-suited to visually-aware conversational recommendation, where systems must efficiently handle both textual dialogue and multiple product images.

7 Conclusion and Discussion

We introduced LaViC, a two-stage method for visually-aware conversational recommendation. First, our *visual knowledge self-distillation* compresses each product image into a small set of [CLS]-positioned embeddings while retaining essential visual details. Second, our *recommendation fine-tuning* enables a large vision-language model to integrate these compressed image embeddings with user dialogues in a unified generative framework. Experiments on the newly collected *Reddit-Amazon* dataset show substantial performance gains over text-only methods, existing vision-language baselines, and even some proprietary systems.

Future work can explore larger or more diverse datasets, where cross-domain attributes may strengthen generalization. Meanwhile, we considered each product's *single* representative image (split into sub-images). Although sufficient in many cases, real-world listings often contain multiple images highlighting different features, suggesting further research on managing richer visual contexts. Our method follows a candidate-based pipeline, meaning its accuracy partly depends on the retrieval module. Enhancing retrieval could further boost recommendation performance. Lastly, this work used a 7B-parameter backbone for computational feasibility. Larger models (e.g., 13B or 34B) could potentially yield stronger multimodal reasoning at the cost of increased inference time. Exploring these trade-offs remains an intriguing direction for future efforts.

References

[1] Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. 2022. VLMo: Unified Vision-Language Pre-Training with Mixture-of-Modality-Experts. In *NeurIPS*.

[2] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott M. Lundberg, Harsha Nori, Hamid Palangi, Marco Túlio Ribeiro, and Yi Zhang. 2023. Sparks of Artificial General Intelligence: Early experiments with GPT-4. *CoRR* (2023).

[3] Li Chen and Pearl Pu. 2012. Critiquing-based recommenders: survey and emerging trends. *User Model. User Adapt. Interact.* (2012).

[4] Qibin Chen, Junyang Lin, Yichang Zhang, Ming Ding, Yukuo Cen, Hongxia Yang, and Jie Tang. 2019. Towards Knowledge-Based Recommender Dialog System. In *EMNLP-IJCNLP*.

[5] Xu Chen, Hanxiong Chen, Hongteng Xu, Yongfeng Zhang, Yixin Cao, Zheng Qin, and Hongyuan Zha. 2019. Personalized Fashion Recommendation with Visual Explanations based on Multimodal Attention Network: Towards Visually Explainable Recommendation. In *SIGIR*.

[6] 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. https://lmsys.org/blog/2023-03-30-vicuna/

[7] Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016. Towards Conversational Recommender Systems. In *KDD*.

[8] Jesse Dodge, Andreea Gane, Xiang Zhang, Antoine Bordes, Sumit Chopra, Alexander H. Miller, Arthur Szlam, and Jason Weston. 2016. Evaluating Prerequisite Qualities for Learning End-to-End Dialog Systems. In *ICLR*.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *ICLR*.

[10] Yue Feng, Shuchang Liu, Zhenghai Xue, Qingpeng Cai, Lantao Hu, Peng Jiang, Kun Gai, and Fei Sun. 2023. A Large Language Model Enhanced Conversational Recommender System. *CoRR* (2023).

[11] Luke Friedman, Sameer Ahuja, David Allen, Zhenning Tan, Hakim Sidahmed, Changbo Long, Jun Xie, Gabriel Schubiner, Ajay Patel, Harsh Lara, Brian Chu, Zexi Chen, and Manoj Tiwari. 2023. Leveraging Large Language Models in Conversational Recommender Systems. *CoRR* (2023).

[12] Zuohui Fu, Yikun Xian, Yaxin Zhu, Shuyuan Xu, Zelong Li, Gerard de Melo, and Yongfeng Zhang. 2021. HOOPS: Human-in-the-Loop Graph Reasoning for Conversational Recommendation. In *SIGIR*.

[13] Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *EMNLP*.

[14] Shirley Anugrah Hayati, Dongyeop Kang, Qingxiaoyang Zhu, Weiyan Shi, and Zhou Yu. 2020. INSPIRED: Toward Sociable Recommendation Dialog Systems. In *EMNLP*.

[15] Ruining He and Julian J. McAuley. 2016. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *WWW*.

[16] Ruining He and Julian J. McAuley. 2016. VBPR: Visual Bayesian Personalized Ranking from Implicit Feedback. In *AAAI*.

[17] Zhankui He, Zhouhang Xie, Rahul Jha, Harald Steck, Dawen Liang, Yesu Feng, Bodhisattwa Prasad Majumder, Nathan Kallus, and Julian J. McAuley. 2023. Large Language Models as Zero-Shot Conversational Recommenders. In *CIKM*.

[18] Zhankui He, Zhouhang Xie, Harald Steck, Dawen Liang, Rahul Jha, Nathan Kallus, and Julian McAuley. 2025. Reindex-then-adapt: Improving large language models for conversational recommendation. In *WSDM*.

[19] Zhankui He, Handong Zhao, Tong Yu, Sungchul Kim, Fan Du, and Julian J. McAuley. 2022. Bundle MCR: Towards Conversational Bundle Recommendation. In *RecSys*.

[20] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian J. McAuley. 2024. Bridging Language and Items for Retrieval and Recommendation. *CoRR* (2024).

[21] 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 *ICLR*.

[22] Meihuizi Jia, Ruixue Liu, Peiying Wang, Yang Song, Zexi Xi, Haobin Li, Xin Shen, Meng Chen, Jinhui Pang, and Xiaodong He. 2022. E-ConvRec: A Large-Scale Conversational Recommendation Dataset for E-Commerce Customer Service. In *LREC*.

[23] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. *CoRR* (2023).

[24] Dongyeop Kang, Anusha Balakrishnan, Pararth Shah, Paul A. Crook, Y-Lan Boureau, and Jason Weston. 2019. Recommendation as a Communication Game: Self-Supervised Bot-Play for Goal-oriented Dialogue. In *EMNLP-IJCNLP*.

[25] Wang-Cheng Kang, Chen Fang, Zhaowen Wang, and Julian J. McAuley. 2017. Visually-Aware Fashion Recommendation and Design with Generative Image Models. In *ICDM*.

[26] Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, Lichan Hong, Ed H. Chi, and Derek Zhiyuan Cheng. 2023. Do LLMs Understand User Preferences? Evaluating LLMs On User Rating Prediction. *CoRR* (2023).

[27] Wenqiang Lei, Xiangnan He, Yisong Miao, Qingyun Wu, Richang Hong, Min-Yen Kan, and Tat-Seng Chua. 2020. Estimation-Action-Reflection: Towards Deep Interaction Between Conversational and Recommender Systems. In *WSDM*.

[28] Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. 2020. Interactive Path Reasoning on Graph for Conversational Recommendation. In *KDD*.

[29] Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. 2018. Towards Deep Conversational Recommendations. In *NeurIPS*.

[30] Shuyang Li, Bodhisattwa Prasad Majumder, and Julian J. McAuley. 2021. Self-Supervised Bot Play for Conversational Recommendation with Justifications. *CoRR* (2021).

[31] Jiayi Liao, Sihang Li, Zhengyi Yang, Jiancan Wu, Yancheng Yuan, Xiang Wang, and Xiangnan He. 2024. LLaRA: Large Language-Recommendation Assistant. In *SIGIR*.

[32] Lizi Liao, Le Hong Long, Zheng Zhang, Minlie Huang, and Tat-Seng Chua. 2021. MMConv: An Environment for Multimodal Conversational Search across Multiple Domains. In *SIGIR*.

[33] Fan Liu, Zhiyong Cheng, Changchang Sun, Yinglong Wang, Liqiang Nie, and Mohan S. Kankanhalli. 2019. User Diverse Preference Modeling by Multimodal Attentive Metric Learning. In *MM*.

[34] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved Baselines with Visual Instruction Tuning. In *CVPR*.

[35] Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. https://llava-vl.github.io/blog/2024-01-30-llava-next/

[36] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual Instruction Tuning. In *NeurIPS*.

[37] Qidong Liu, Jiaxi Hu, Yutian Xiao, Xiangyu Zhao, Jingtong Gao, Wanyu Wang, Qing Li, and Jiliang Tang. 2025. Multimodal Recommender Systems: A Survey. *ACM Comput. Surv.* (2025).

[38] Qiang Liu, Shu Wu, and Liang Wang. 2017. DeepStyle: Learning User Preferences for Visual Recommendation. In *SIGIR*.

[39] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR* (2019).

[40] Yuqing Liu, Yu Wang, Lichao Sun, and Philip S. Yu. 2024. Rec-GPT4V: Multimodal Recommendation with Large Vision-Language Models. *CoRR* (2024).

[41] Yuanxing Liu, Weinan Zhang, Baohua Dong, Yan Fan, Hang Wang, Fan Feng, Yifan Chen, Ziyu Zhuang, Hengbin Cui, Yongbin Li, and Wanxiang Che. 2023. U-NEED: A Fine-grained Dataset for User Needs-Centric E-commerce Conversational Recommendation. In *SIGIR*.

[42] Zeming Liu, Haifeng Wang, Zhengyu Niu, Hua Wu, and Wanxiang Che. 2021. DuRecDial 2.0: A Bilingual Parallel Corpus for Conversational Recommendation. In *EMNLP*.

[43] Srijata Maji, Moghis Fereidouni, Vinaik Chhetri, Umar Farooq, and A. B. Siddique. 2024. MobileConvRec: A Conversational Dataset for Mobile Apps Recommendations. *CoRR* (2024).

[44] Julian J. McAuley, Christopher Targett, Qinfeng Shi, and Anton van den Hengel. 2015. Image-Based Recommendations on Styles and Substitutes. In *SIGIR*.

[45] Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. 2019. OpenDialKG: Explainable Conversational Reasoning with Attention-based Walks over Knowledge Graphs. In *ACL*.

[46] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *ICML*.

[47] Filip Radlinski, Krisztian Balog, Bill Byrne, and Karthik Krishnamoorthi. 2019. Coached Conversational Preference Elicitation: A Case Study in Understanding Movie Preferences. In *SIGdial*.

[48] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *EMNLP-IJCNLP*.

[49] Stephen E. Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.* (2009).

[50] John Schulman, Barret Zoph, C Kim, Jacob Hilton, Jacob Menick, Jiayi Weng, Juan Felipe Ceron Uribe, Liam Fedus, Luke Metz, Michael Pokorny, Rapha Gontijo Lopes, and Sengjia Zhao. 2022. Chatgpt: Optimizing language models for dialogue. *OpenAI* (2022).

[51] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. LLaMA: Open and Efficient Foundation Language Models. *CoRR* (2023).

[52] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *CoRR* (2017).

[53] Qifan Wang, Yinwei Wei, Jianhua Yin, Jianlong Wu, Xuemeng Song, and Liqiang Nie. 2023. DualGNN: Dual Graph Neural Network for Multimedia Recommendation. *IEEE Trans. Multim.* (2023).

[54] Suhang Wang, Yilin Wang, Jiliang Tang, Kai Shu, Suhas Ranganath, and Huan Liu. 2017. What Your Images Reveal: Exploiting Visual Contents for Point-of-Interest Recommendation. In *WWW*.

[55] Xiaolei Wang, Xinyu Tang, Xin Zhao, Jingyuan Wang, and Ji-Rong Wen. 2023. Rethinking the Evaluation for Conversational Recommendation in the Era of Large Language Models. In *EMNLP*.

[56] Xiaolei Wang, Kun Zhou, Ji-Rong Wen, and Wayne Xin Zhao. 2022. Towards Unified Conversational Recommender Systems via Knowledge-Enhanced Prompt Learning. In *KDD*.

[57] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019. MMGCN: Multi-modal Graph Convolution Network for Personalized Recommendation of Micro-video. In *MM*.

[58] Ga Wu, Kai Luo, Scott Sanner, and Harold Soh. 2019. Deep language-based critiquing for recommender systems. In *RecSys*.

[59] Zhouhang Xie, Junda Wu, Hyunsik Jeon, Zhankui He, Harald Steck, Rahul Jha, Dawen Liang, Nathan Kallus, and Julian J. McAuley. 2024. Neighborhood-Based Collaborative Filtering for Conversational Recommendation. In *RecSys*.

[60] Hu Xu, Seungwhan Moon, Honglei Liu, Bing Liu, Pararth Shah, and Philip S. Yu. 2020. User Memory Reasoning for Conversational Recommendation. In *COLING*.

[61] Zhengyuan Yang, Linjie Li, Kevin Lin, Jianfeng Wang, Chung-Ching Lin, Zicheng Liu, and Lijuan Wang. 2023. The Dawn of LMMs: Preliminary Explorations with GPT-4V(ision). *CoRR* (2023).

[62] Zixuan Yi, Zijun Long, Iadh Ounis, Craig Macdonald, and Richard McCreadie. 2023. Large Multi-modal Encoders for Recommendation. *CoRR* (2023).

[63] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid Loss for Language Image Pre-Training. In *ICCV*.

[64] Jinghao Zhang, Yanqiao Zhu, Qiang Liu, Shu Wu, Shuhui Wang, and Liang Wang. 2021. Mining Latent Structures for Multimedia Recommendation. In *MM*.

[65] Yiming Zhang, Lingfei Wu, Qi Shen, Yitong Pang, Zhihua Wei, Fangli Xu, Bo Long, and Jian Pei. 2022. Multiple Choice Questions based Multi-Interest Policy Learning for Conversational Recommendation. In *WWW*.

[66] Kun Zhou, Wayne Xin Zhao, Shuqing Bian, Yuanhang Zhou, Ji-Rong Wen, and Jingsong Yu. 2020. Improving Conversational Recommender Systems via Knowledge Graph based Semantic Fusion. In *KDD*.

[67] Kun Zhou, Yuanhang Zhou, Wayne Xin Zhao, Xiaoke Wang, and Ji-Rong Wen. 2020. Towards Topic-Guided Conversational Recommender System. In *COLING*.

[68] Jie Zou, Evangelos Kanoulas, Pengjie Ren, Zhaochun Ren, Aixin Sun, and Cheng Long. 2022. Improving Conversational Recommender Systems via Transformer-based Sequential Modelling. In *SIGIR*.

A Prompts

Our data collection and model training involve several prompt templates, summarized below. In particular, Table 9 shows the prompt used to filter Reddit conversations for recommendation requests, while Table 10 describes how we generate product-specific image descriptions for visual knowledge self-distillation. Table 11 illustrates the prompts for the generative recommendation task in both text-only and text & image settings.

Filtering recommendation conversations. We use GPT-3.5-turbo with a simple instruction (Table 9) to identify whether the last utterance in a Reddit thread contains an explicit product recommendation. Only conversations labeled as 'yes' are retained in our *Reddit-Amazon* dataset.

Generating image descriptions. For visual knowledge self-distillation (Section 4.2), we guide the model to produce a visually grounded description of each product (Table 10). The prompt encourages the model to focus on appearance and features visible in the product image without simply restating the product title.

Recommendation task with or without images. Table 11 displays the prompt template for text-only and visually enhanced recommendation. The prompt asks the model to read a conversation and a list of candidate products, then recommend the most relevant item by returning its ID. In the image-based version, each candidate includes an additional image input for multimodal reasoning.

Table 9: Prompt for filtering recommendation-related conversations. Our *Reddit-Amazon* dataset includes only conversations that GPT-3.5-turbo [50] answered 'yes'.

> You are a helpful assistant. I will show you a conversation on Reddit. Each utterance is included in a tag <utterance>...</utterance> and numbered as (1), (2), (3), etc (i.e., the format is <utterance>(i) [content]</utterance>). The goal is to determine whether the final utterance in the conversation is recommending an item. If any of the preceding utterances involve requesting an item recommendation, and the last utterance explicitly recommends one or more items, classify the conversation as a recommendation. Please answer 'yes' if the last utterance is a recommendation of items, otherwise please answer 'no'. Please answer 'uncertain' if you cannot classify 'yes' or 'no' from the given conversation. Do not include other phrases in your answer.
>
> Conversation: **{conversation}**

Table 10: Prompt for generating image description.

> You are a helpful AI assistant. Given an Amazon product's title and its image, please provide a detailed, visually grounded description of the product that would help someone decide whether to purchase it. Please focus on the product's appearance, features, and any other visually informative aspects. Do not mention the product's title in your answer.
>
> This product's title is: **{title}**
> **{image}**

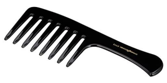
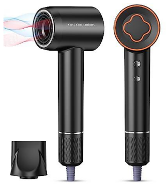
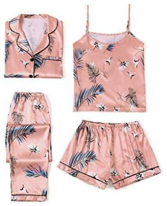
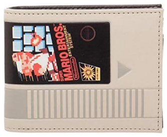
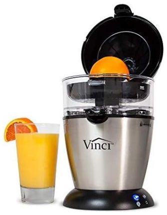
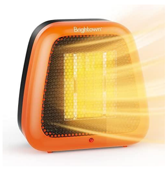
Table 11: Prompt for recommendation used by generative baselines and LaViC. *Gray italics* indicate phrases that are used only when images are provided.

> You are an AI assistant specialized in providing personalized product recommendations based on user conversations. You are given a conversation between a user seeking recommendation (denoted by ⟨submission⟩) and other users providing comments (denoted by ⟨comment⟩). You are also given a set of candidate products with their IDs, titles*, and images* formatted as "ID: title"*followed by an image*. Among the candidates, recommend the most relevant product to the seeker. Only reply with its ID, and don't say anything else.
>
> Conversation: **{conversation}**
>
> Candidates: **{list of candidates}**

B Examples of Image Descriptions from LLaVA-v1.6

Tables 12 shows examples of image descriptions generated by LLaVA-v1.6 in our visual knowledge self-distillation step. These examples are drawn from three domains in the *Reddit-Amazon* dataset: *beauty*, *fashion*, and *home*. For each product, we provide its image, domain label, product title, and the description automatically generated by LLaVA-v1.6. We highlight in yellow shading specific visual details inferred from the image, such as color, material, design elements, or functional components. These features are not explicitly mentioned in the product title, demonstrating the model's capacity to generate visually grounded information. In our *visual knowledge self-distillation* process (Section 4.2), we distill only a small set of tokens to capture this descriptive knowledge for subsequent recommendation tasks. This approach preserves the essential details found in the images while mitigating token overflows when multiple candidates must be processed simultaneously.

Table 12: Examples of image descriptions generated by LLaVA-v1.6 using full tokens. Highlighted text indicates additional visual details inferred from the image that are not explicitly mentioned in the product title.

Image	Domain	Title	Description generated by LLaVA-v1.6
	Beauty	Hercules Sägemann, Magic Star comb, 9 inches, 1 set, (1 x 1 piece), 5660	The product in the image is a wide-tooth comb, specifically the Hercules Sägemann Magic Star comb. It is 9 inches in length and appears to be made of a durable material, likely plastic or a similar synthetic. The comb has a sleek, black design with wide, evenly spaced teeth, making it suitable for detangling hair without causing damage. The handle is ergonomically designed for a comfortable grip, and the comb features the brand name “Hercules Sägemann” printed on it. This type of comb is ideal for use on wet or dry hair and is particularly useful for individuals with curly or thick hair.
	Beauty	Hair Dryer - 150000 RPM High-Speed Brushless Motor Negative Ionic Blow Dryer for Fast Drying, Low Noise Thermo-Control Hair Dryers with Magnetic Nozzle, Perfect for Gifts	Based on the image, the product is a sleek, modern hair dryer with a high-speed brushless motor. The hair dryer has a cylindrical design with a matte black finish and silver accents. It features a magnetic nozzle attachment for focused airflow, which can be seen detached in the image. The front of the hair dryer shows the air outlet, which appears to be designed for efficient airflow, with blue and red lines indicating the flow of air. The handle of the hair dryer has a textured grip for easy handling and two buttons, likely for controlling the speed and heat settings. The back of the hair dryer has a unique design with an orange ring and a cross-shaped pattern, possibly indicating the air intake area. Overall, the hair dryer looks compact and stylish, designed for fast drying with low noise, and it includes advanced features like negative ionic technology and thermo-control. The product appears to be suitable for gifting, as suggested by its elegant design and high-tech features.
	Fashion	LYANER Women's Pajamas Set 4pcs Satin Silk Cami Top Button Down Loungewear Pjs Set	The product is a women's pajama set made of satin silk, consisting of four pieces. The set includes: 1. A button-down long-sleeve top with a collar and black piping details. 2. A cami top with thin spaghetti straps. 3. A pair of long pants with an elastic waistband. 4. A pair of shorts with an elastic waistband. The fabric has a smooth, shiny finish and features a pink base color with a tropical print, including leaves and flowers in shades of blue, white, and black. The overall design is elegant and comfortable, suitable for lounging and sleeping.
	Fashion	Super Mario Bros Classic Retro NES Design Bifold Wallet	The product is a bi-fold wallet designed with a retro theme inspired by the classic Super Mario Bros game for the NES (Nintendo Entertainment System). The wallet features a design that mimics the appearance of an NES game cartridge, with the iconic Super Mario Bros label prominently displayed on the front. The label includes pixelated graphics reminiscent of the original game, along with the Nintendo logo and other classic design elements. The wallet appears to be made of a material that resembles the texture and color of an NES cartridge, giving it a nostalgic and vintage look.
	Home	Vinci Hands-Free Patented Electric Citrus Juicer 1-Button Easy Press Lemon Lime Orange Grapefruit Juice Squeezer Easy to Clean Juicer Machine, Black/Stainless Steel	The product in the image is an electric citrus juicer by Vinci. It features a sleek design with a combination of black and stainless steel colors. The juicer has a transparent upper section where the citrus fruit is placed, and it appears to be in the process of juicing an orange. The machine has a one-button operation for easy use, as indicated by the buttons on the base. The base is black and has a sturdy appearance. Next to the juicer, there is a glass filled with freshly squeezed orange juice, garnished with a slice of orange, showcasing the juicer's capability to produce fresh juice. The overall design suggests that the juicer is easy to clean and user-friendly.
	Home	Small Space Heater for Indoor Use - 400W Low Wattage Mini Heater with Overheat Protection, Portable Personal Heater with Tip Over Protection, Low Noise Desk Heater for Office Bedroom Home Use	Based on the image, the product is a compact, portable space heater designed for indoor use. It has a modern, triangular design with an orange front and a black back. The front features a mesh grille through which the heating elements are visible, and it appears to be emitting warm air. The heater is branded with the name B̈rightownät the top. It is likely designed for personal use in small spaces such as an office, bedroom, or home. The heater includes safety features such as overheat protection and tip-over protection, and it operates with low noise, making it suitable for quiet environments.