Navigating User Behavior toward Personalized Multimodal Generation

Anonymous ACL submission

Abstract

Modern AIGC pipelines deliver high-fidelity images and videos but presuppose a well-formed creation instruction, while end users rarely articulate visual details, leaving generators misaligned with user demand. We study **personalized content generation**, which turns a user's interaction history into an executable instruction for downstream synthesis, and identify two obstacles: behavior must be encoded in a form legible to language reasoning, and the model must acquire instruction-writing skill absent from both pretraining and behavior data. We propose **NaviGen**, which represents each item with a dual identifier coupling a collaborative code and a textual code as a behavioral substrate and a semantic bridge in one token stream. On this representation, a two-stage SFT+RL pipeline first distills preference reasoning and instruction writing from evolutionarily searched supervision, then aligns generation with user intent through hierarchical and self-consistent rewards. Experiments across product, game, and short-video domains show that NaviGen improves personalized image and video generation, strengthens next-item prediction, and yields more specific, relevant, and visually generatable instructions. Our code is anonymously released at: https://anonymous.4open.science/r/NaviGen-08F8/.

1 Introduction

Multimodal content generation, such as text-to-image posters and short videos, is rapidly becoming a core productivity layer for media, marketing, and e-commerce (Xu et al., 2025; Ling et al., 2026). A modern creation pipeline refines a textual instruction with a language model and renders it into an image or video via a text-to-vision generator (Yang et al., 2025; Seedance et al., 2026), where the instruction serves as the central control signal of what is generated and how it looks.

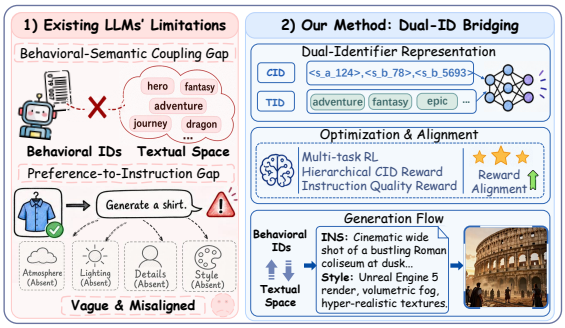


Figure 1: Personalized multimodal content generation.

Recent progress along this pipeline broadly falls into three lines. *(i) Generation backbones:* diffusion and autoregressive transformers for text-to-image and text-to-video deliver high-fidelity rendering from textual prompts (Yang et al., 2025; Seedance et al., 2026). *(ii) Instruction enrichment:* LLM-based prompt expansion and multi-agent creation systems turn short user inputs into detailed, structured instructions that better exploit generator capacity (Xu et al., 2025; An et al., 2026; Dang et al., 2026). *(iii) Conditional control:* reference-, layout-, or identity-conditioned generation injects external signals for fine-grained control over the output (Zhao et al., 2025b; Ling et al., 2026).

However, all of these methods presuppose **a well-formed creation instruction as input**, leaving a fundamental question unanswered: *whose taste is the content actually for?* End consumers differ widely in taste and rarely articulate visual details, yet whether the content resonates with them is what ultimately matters. Without a path from user signals to a concrete instruction, even the strongest generators produce content that is generic or misaligned with real demand. This gap motivates a new problem we term ***personalized content generation***: translating a user's implicit behavior history into a creation instruction that steers the downstream generator toward content the user truly wants.

Realizing this paradigm raises two central chal-

lenges, on the input and output sides of the language model respectively. **(C1) Representation gap between behavior and language.** A user's behavior history must be encoded into a form the LM can reason over, yet no single representation suffices: ID-based encodings preserve behavioral structure but remain opaque in the LM's semantic space, while raw textual metadata is expressive but verbose, tempting the model to paraphrase history rather than infer preference. A workable representation must carry behavioral signal and stay legible to language reasoning at once. **(C2) Capability gap between understanding preference and writing instructions.** Even given such a representation, *knowing what a user likes* and *writing a good creation instruction* are two distinct skills—the latter neither cultivated by language pretraining nor reflected in user behavior data, leaving the model with no natural source to acquire it.

We propose NaviGen, whose two core designs map one-to-one onto these challenges. To close the representation gap (C1), NaviGen encodes each item with a **dual-identifier** scheme: a collaborative identifier (CID) captures its behavioral role via residual vector quantization, while a textual identifier (TID) compresses its textual semantics into ordered, standardized terms. Together they give the language model a compact behavioral substrate and a controllable language bridge in a single token stream. To close the capability gap (C2), NaviGen adopts a **two-stage SFT+RL pipeline**. The SFT stage learns from history-to-instruction supervision synthesized by evolutionary search under an LLM judge, teaching the model to reason about preference evolution rather than paraphrase history. The RL stage jointly optimizes two complementary rewards: a hierarchical CID reward for preference correctness, and a triangular instruction-aware reward defined over the generated instruction, the predicted target semantics, and the ground-truth target semantics, which together drive the model toward generation-ready instructions.

Our contributions are summarized as follows:

- We propose NaviGen, a unified framework that turns user behavior sequences into generation-ready creation instructions for personalized AIGC, bridging behavioral modeling and controllable content generation in a single pipeline.
- We introduce a dual-identifier representation that couples a residual-quantized CID with an ordered, length-flexible TID, jointly providing a compact behavioral substrate and a controllable semantic bridge within one token stream.
- We design a two-stage SFT+RL pipeline that requires no human-written instructions: evolutionary search with an LLM judge synthesizes supervision, while GRPO uses a hierarchical CID reward and a triangular instruction-aware reward enforcing closed-loop self-consistency.
- Across product, game, and short-video domains, NaviGen consistently improves personalized image and video generation quality, CID-space next-item prediction accuracy, and instruction specificity, relevance, and visual generatability.

2 Preliminary

Personalized Content Generation. We consider the setting of consumer-facing personalized AIGC, where a multimodal generative model synthesizes visual content tailored to an individual user. Given a textual creation instruction $\mathcal{I} \in \mathcal{T}$, an off-the-shelf generator g_ϕ produces the final output:

$$\mathcal{O} \;=\; g_\phi(\mathcal{I}), \tag{1}$$

where $\mathcal{O} \in \mathcal{Y}$ denotes the generated content in the target modality space (*e.g.*, image or video), and g_ϕ remains fixed during our training. Under this setting, the quality of personalized generation is fundamentally bottlenecked by the quality of $\mathcal{I}$, while end consumers cannot be expected to author such instructions by hand. This motivates the need for an automatic instruction generator f_θ that produces $\mathcal{I}$ on the user's behalf.

Behavior as Implicit Preference Evidence. To drive f_θ toward user-specific generation, we leverage the user's observed interaction history as an implicit signal of visual preference. For a given user, we denote this history as an ordered sequence

$$\mathcal{H}_u \;=\; \langle x_1, x_2, \ldots, x_n \rangle, \tag{2}$$

where each x_k is an item the user has previously engaged with (*e.g.*, clicked, viewed, or purchased), associated with its visual and semantic attributes. We treat $\mathcal{H}_u$ as *preference evidence*: a record from which the user's latent visual taste can be inferred and projected forward into a creative direction.

Task Formulation: Behavior-Conditioned Instruction Generation. Given a user's history $\mathcal{H}_u$, our goal is to learn an instruction generator f_θ that produces a free-form textual instruction:

$$\mathcal{I} \;=\; f_\theta(\mathcal{H}_u), \quad \max_\theta \; P_\theta(\mathcal{I} \mid \mathcal{H}_u)\,. \tag{3}$$

We say that an instruction $\mathcal{I}$ is *generation-ready* if it satisfies two properties: (1) *preference alignment*: $\mathcal{I}$ faithfully captures the user-specific visual preferences evidenced by $\mathcal{H}_u$; and (2) *generation feasibility*: $\mathcal{I}$ is sufficiently concrete and visually grounded to serve as an effective conditioning signal for the downstream generator g_ϕ. The objective of this work is to design f_θ such that, for arbitrary user histories, it consistently emits generation-ready instructions, thereby bridging implicit user behavior and high-quality personalized visual synthesis.

3 Method

This section presents the technical details of NaviGen, with overall architecture shown in Figure 2.

3.1 Dual-Identifier Behavior Encoding

To make user behavior $\mathcal{H}_u$ legible to an LLM, each entry $x_k \in \mathcal{H}_u$ must be serialized into tokens within the model's vocabulary. A naive choice, directly feeding captions or metadata, is verbose and injects redundant noise that slows optimization and blurs preference signals. We therefore encode each entry with a compact *dual identifier*, decoupling sequence-level identity from semantic grounding so that neither role compromises the other.

Collaborative Identifier (CID). Inspired by LLMs for collaborative filtering (Deng et al., 2025), the CID encodes an entry's role within user interaction sequences, distilling collaborative patterns observed across the consumer-content interaction graph. Its metadata m_v is first mapped to a continuous embedding $\mathbf{e}_v = \psi(m_v)$ via a pretrained embedding model ψ, then quantized through a multi-layer residual K-means process:

$$s_\ell = \arg\min_k \|\mathbf{r}_\ell - \mathbf{c}_\ell^k\|^2, \ \mathbf{r}_{\ell+1} = \mathbf{r}_\ell - \mathbf{c}_\ell^{s_\ell}, \quad (4)$$

where $\mathbf{r}_1 = \mathbf{e}_v$, $\mathbf{c}_\ell^k$ denotes the k-th centroid in the ℓ-th codebook, and $s_\ell \in \{1, \ldots, K_{\text{cb}}\}$ is the discrete code assigned to level ℓ. The resulting CID is a three-level residual token sequence:

$$\text{cid}(v) = \langle\ s_1(v),\ s_2(v),\ s_3(v)\ \rangle. \quad (5)$$

This hierarchy enables multi-granularity modeling and partial-credit supervision, as matching any level yields a meaningful signal. Each CID token is added to the vocabulary and initialized via dedicated embedding training (Section 3.2.1).

Textual Identifier (TID). Unlike existing work that treats TIDs as fixed-length targets for next-item prediction (Zhang et al., 2026), we note that semantically equivalent textual variants may correspond to different terms, making exact next-TID prediction overly restrictive; meanwhile, entries vary in semantic complexity and thus require different numbers of terms. We therefore construct variable-length TIDs by imposing only an upper bound on the number of terms:

$$\text{tid}(v) = [t_1, t_2, \ldots, t_m], \quad m \leq 10, \quad (6)$$

where each t_k is a concise phrase capturing a core semantic dimension (*e.g.*, subject category, key attribute). Terms are ordered by importance and produced by compressing the original caption through an LLM with controlled output constraints. Unlike free-form text, the TID provides a compact, deduplicated, and domain-stable semantic signature that serves as the bridge between sequence-level preference modeling and instruction generation.

3.2 Reasoning-Infused Supervised Tuning

NaviGen employs two-stage supervised fine-tuning that progressively builds from identifier embeddings to full reasoning-capable generation.

3.2.1 Cold-Start Embedding Initialization

To prevent randomly initialized CID embeddings from destabilizing pretrained weights through noisy gradients, we decouple representation acquisition from backbone adaptation: all pretrained weights are frozen, and only parameters tied to the new tokens are updated. Let $\mathcal{D}_{\text{init}}$ denote the auxiliary training set with token sequences $\mathbf{s}_i$ and target tokens y_i. The embedding initialization loss is:

$$\mathcal{L}_{\text{init}} = -\sum_i \sum_t \log p(y_{i,t} \mid \mathbf{s}_{i,<t}; \mathcal{E}_{\text{CID}}, \mathbf{W}_{\text{out}}) \quad (7)$$

where $\mathcal{E}_{\text{CID}}$ and $\mathbf{W}_{\text{out}}$ are the learnable CID embeddings and output projection layer. Three auxiliary tasks establish bidirectional CID-TID alignment:

CID2TID. Mapping CID to its corresponding TID, grounding behavioral codes in semantic terms.

TID2CID. Inverse TID-to-CID mapping, constructing behavioral identifiers from semantic signals.

CID2CID. Predicting a target CID from a history of CIDs, capturing sequential behavioral patterns.

3.2.2 Reasoning-Augmented Full Finetuning

The initialization stage equips the model with stable CIDs, but two capabilities essential for behavior-conditioned instruction generation remain absent: (i) translating inferred preferences into generation-ready instructions, and (ii) reasoning

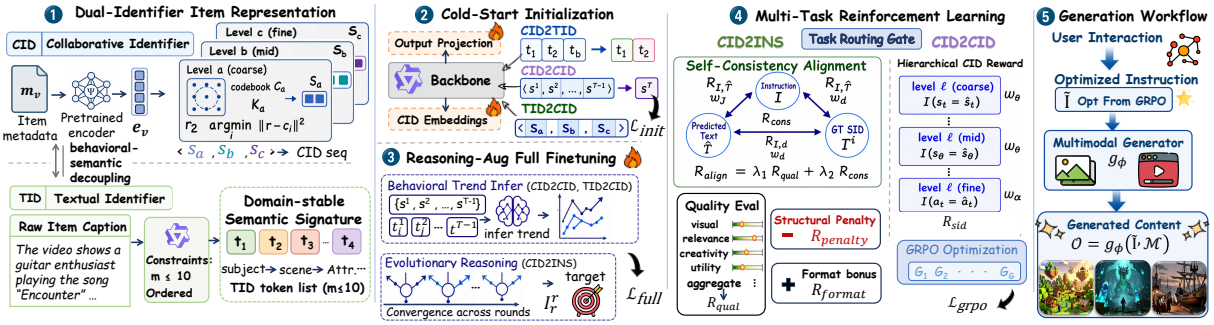


Figure 2: Overall architecture of the proposed NaviGen framework for personalized multimodal generation.

over how user interests evolve along the interaction history. To instill both, we unfreeze all parameters and augment the existing objectives with a new **CID2INS** task and chain-of-thought supervision.

CID2INS. AIGC instructions are synthesized through an evolution-inspired search that progressively refines candidate instructions toward user-aligned visual semantics. Starting from three founder strategies (conservative, balanced, exploratory), each round selects the two strongest candidates via a multi-dimensional judge and produces two offspring through crossover and controlled mutation. Given the candidate trajectory $\{\mathcal{I}^{(r)}\}_{r=0}^{R}$, the final instruction is:

$$\mathcal{I}^{\star} = \arg\max_{\mathcal{I}\in\mathcal{P}_{\mathrm{f}}} \mathcal{S}_{\mathrm{judge}}(\mathcal{I}), \tag{8}$$

where $\mathcal{P}_{\mathrm{f}}$ is the final population and $\mathcal{S}_{\mathrm{judge}}$ is the LLM-based multi-dimensional scorer. The model is jointly supervised to predict the target TID, anchoring the generated instruction to the intended visual semantics and preventing semantic drift.

Reasoning. Chain-of-thought traces are distilled from a teacher model. For **CID2CID**, the teacher articulates how user preferences evolve along the historical identifier sequence without explicitly referencing the target item. Formally, given an identifier sequence $h_{1:n} = (h_1, \ldots, h_n)$ where h_i denotes the CID at step i, the teacher distills consecutive preference shifts into a reasoning chain:

$$\mathcal{T}\big(\{h_{i-1} \rightarrow h_i\}_{i=2}^{n}\big), \tag{9}$$

where $\mathcal{T}(\cdot)$ aggregates the evolving preference trajectory across the interaction history. For **CID2INS**, the reasoning summarizes the full evolutionary trajectory—how candidate instructions converged toward the target semantics across rounds. Formally, starting from founder strategies that form the initial population $\mathcal{P}^{(0)}$, each round $r \in \{1, \ldots, R\}$ selects the two highest-scoring candidates under $\mathcal{S}_{\mathrm{judge}}$ and updates the population via elitism, crossover, and mutation:

$$\mathcal{P}^{(r)} = \big\{\, \mathcal{I}^{(r)}_{\mathrm{elite}},\ \mathcal{I}^{(r)}_{\mathrm{elite}} \oplus \mathcal{I}^{(r)}_{\mathrm{mate}},\ \mu(\mathcal{I}^{(r)}_{\mathrm{elite}}) \,\big\}, \tag{10}$$

where $\mathcal{I}^{(r)}_{\mathrm{elite}}$ and $\mathcal{I}^{(r)}_{\mathrm{mate}}$ are the top-2 candidates in $\mathcal{P}^{(r-1)}$, $\oplus$ fuses the elite's target-facing anchor with the mate's visual expressiveness, and $\mu(\cdot)$ applies a controlled mutation that preserves the semantic core. The resulting score trajectory across rounds teaches the model to connect evolutionary search dynamics with the final output.

Let $\mathcal{D}_{\mathrm{full}}$ denote the reasoning-augmented training set with input sequences $\mathbf{s}_i$ and output sequences $\mathbf{y}_i$ containing reasoning traces and structured answers. The full finetuning objective is:

$$\mathcal{L}_{\mathrm{full}} = -\sum_{i}\sum_{t} \log p(\mathbf{y}_{i,t} \mid \mathbf{s}_{i,<t}; \theta), \tag{11}$$

where $p(\cdot \mid \mathbf{s}_{i,<t}; \theta)$ denotes the token-level probability under all trainable parameters θ. The resulting model jointly performs preference-grounded reasoning and behavior-conditioned instruction generation for downstream synthesis.

3.3 Multi-Task Reinforcement Learning

Supervised fine-tuning produces a competent generator but does not directly optimize for the quality of personalized outputs. NaviGen applies GRPO to refine the policy π_θ under task-specific reward signals that better reflect downstream objectives.

Hierarchical CID Reward. The CID encodes item behavior through three residual levels of decreasing granularity. A match at any level provides a meaningful signal, but coarse agreement is weighted more heavily than fine precision, reflecting the intuition that predicting the right item family matters more than nailing the exact variant. Formally, given a predicted CID $\hat{s} = (\hat{s}_a, \hat{s}_b, \hat{s}_c)$ and ground truth $\tilde{s} = (\tilde{s}_a, \tilde{s}_b, \tilde{s}_c)$, the hierarchical reward is:

$$R_{\mathrm{cid}} = \sum_{\tau\in\{a,b,c\}} w_\tau \cdot \mathbb{I}[\hat{s}_\tau = \tilde{s}_\tau], \tag{12}$$

Table 1: Statistics of the experimental datasets.

Dataset	#Users	#Items	#Interactions	Density
Product	30161	74487	161405	7.18e-5
Games	25912	35443	144337	1.57e-4
Short Videos	35259	153333	209172	3.87e-5

where w_a, w_b and w_c enforce the coarse-to-fine weighting. A small bonus rewards predictions that remain within the valid CID vocabulary even when they do not match the ground truth, encouraging the model to stay within the learned identifier space.

Instruction-Aware Reward. For **CID2INS**, the reward combines instruction quality assessment with a triangular self-consistency check. An LLM-based judge evaluates the instruction along four dimensions—*specificity*, *creativity*, *content quality*, and *visual generatability*, aggregated into R_{qual}. Beyond standalone quality, a closed-loop alignment enforces three mutually reinforcing signals: the instruction must anchor to target semantics $R_{\text{ins}\leftrightarrow\tilde{t}}$, remain self-consistent with its own prediction $R_{\text{ins}\leftrightarrow\hat{t}}$, and the prediction itself must align with ground truth $R_{\hat{t}\leftrightarrow\tilde{t}}$. The combined reward is:

$$R_{\text{align}} = \gamma_1 R_{\text{ins}\leftrightarrow\tilde{t}} + \gamma_2 R_{\text{ins}\leftrightarrow\hat{t}} + \gamma_3 R_{\hat{t}\leftrightarrow\tilde{t}}, \quad (13)$$

$$R_{\text{ins}} = \lambda_1 \cdot R_{\text{qual}} + \lambda_2 \cdot R_{\text{align}}, \quad (14)$$

where $\gamma_1, \gamma_2, \gamma_3$ balance the three alignment signals, and λ_1, λ_2 weight quality against consistency.

Optimization Objective. We optimize the GRPO objective over a group of G completions:

$$\mathcal{J}_{\text{grpo}} = \mathbb{E}_{\mathbf{q}}[\,\mathcal{L}_{\text{clip}} - \kappa\,\mathbb{D}_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})\,], \quad (15)$$

where $\mathcal{L}_{\text{clip}}$ is the clipped surrogate loss averaged over the group, ρ_i is the importance sampling ratio, ϵ the clipping range, and κ the KL penalty weight. The group-relative advantage $\hat{A}_i$ is driven by a composite reward:

$$R_{\text{task}} = \begin{cases} w_{\text{cid}} R_{\text{cid}} + R_{\text{bonus}}, \\ w_{\text{ins}} R_{\text{ins}} + R_{\text{format}} - R_{\text{penalty}}, \end{cases} \quad (16)$$

where R_{bonus} encourages vocabulary-range adherence, R_{format} verifies JSON parseability and reasoning completeness, and R_{penalty} penalizes structural violations. The optimized instruction $\tilde{\mathcal{I}}$ from this stage then serves as the control signal for the multi-modal generator g_ϕ defined in Section 2.

4 Evaluation

We evaluate NaviGen by answering five research questions. **RQ1** compares personalized AIGC generation performance. **RQ2** studies the contribution of key modules. **RQ3** examines collaborative identifier prediction in the CID space. **RQ4** analyzes the effect of SFT and RL steps. **RQ5** examines qualitative instruction cases.

4.1 Experimental Settings

4.1.1 Datasets and Evaluation Protocols

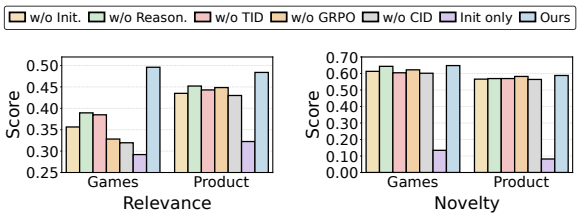
Table 1 reports statistics of the three datasets: **Product** and **Games** are Amazon review domains with captions derived from product metadata, while **Short Videos** comes from the OpenOneRec short-video benchmark, where each item has a textual caption (Zhou et al., 2025). Each data instance is formed by pairing a user's history interaction sequence with a target item, and all instances are split into training, validation, and test sets with an 8:1:1 ratio; all methods are evaluated by Recall@K/NDCG@K. For personalized generation, we sample 1,000 image-generation and 100 video-generation cases from the test split of each dataset, while keeping the AIGC instruction modality consistent across methods. We compare NaviGen and baselines along four dimensions: **Consistency**, measured by image-instruction CLIPScore and, for videos, average CLIPScore over one frame per second (Hessel et al., 2021); **Relevance**, measured by cosine similarity between the generated instruction and the equal-weight embedding of history and target item captions; **Aesthetic**, where a VLM judge scores visual quality in $[0, 1]$ (Kao et al., 2017); and **Novelty**, where the same judge scores novelty and interestingness in $[0, 1]$ (Vargas and Castells, 2011). Details are provided in Appendix A.3

4.1.2 Baseline Methods

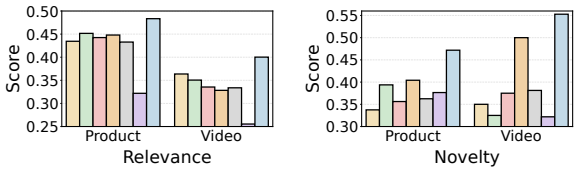
NaviGen is compared with a comprehensive set of baselines, including **i) Personalized Generation Methods:** PMG (Shen et al., 2024), Pigeon (Xu et al., 2025), RAGAR (Ling et al., 2026), CIPHER (Gao et al., 2024), PROSE (Aroca-Ouellette et al., 2025), TRIPLE (Noh et al., 2026); **ii) Collaborative Filtering Methods:** SASRec (Kang and McAuley, 2018), TIGER (Rajput et al., 2023), LC-Rec (Zheng et al., 2024), OpenOneRec-Pretrain (Zhou et al., 2025); and **iii) Prompting and Reference Baselines:** **NPC** removes user-specific evidence and relies only on a generic task prompt, while **Oracle** conditions on the ground-truth target caption or TID as a non-deployable target-conditioned reference.

Table 2: Overall performance comparison on personalized AIGC instruction generation. Aesthetic and Novelty are averaged over three runs. Excluding Oracle, boldface marks the best image-generation result, while boldface with superscript $\star$ marks the best video-generation result.

Method	Output	Games				Product				Short Videos			
		Cons.	Nov.	Aes.	Rel.	Cons.	Nov.	Aes.	Rel.	Cons.	Nov.	Aes.	Rel.
PMG	Image	0.3281	0.5084	0.7424	0.4053	**0.3282**	0.5091	0.7677	0.4044	0.3316	0.5725	0.7376	0.3249
PIGEON	Image	0.3121	0.5449	0.7352	0.4382	0.3104	0.4662	0.7086	0.4347	0.3150	0.5510	0.7291	0.3571
RAGAR	Image	0.3148	0.5002	0.7696	0.4475	0.3122	0.4223	0.7171	0.4204	0.3195	0.5713	0.7410	0.3330
Cipher	Image	0.3213	0.5403	0.7501	0.4273	0.3193	0.4595	0.7362	0.4038	0.3265	0.5936	0.7311	0.3963
	Video	0.6315	0.5245	0.5855	0.4307	0.6104	0.4020	0.4423	0.3890	0.6379	**0.6280***	**0.6830***	0.3595
Prose	Image	0.3173	0.5796	0.7804	0.4052	0.3130	0.4958	0.7398	0.4298	0.3202	0.5936	0.7475	0.3034
	Video	0.6325	0.4740	0.5215	**0.4560***	0.6282	0.4640	0.4185	0.4199	0.6459	0.5355	0.6245	0.3554
TRIPLE	Image	0.3152	0.5082	0.7472	0.4273	0.3093	0.4273	0.7182	0.4038	0.3164	0.5654	0.7236	0.3963
	Video	0.6203	0.4170	0.4290	0.4559	0.6217	0.3510	0.3905	0.4093	0.6419	0.4305	0.5330	0.3315
NPC	Image	0.3238	0.5535	0.7797	0.3728	0.3188	0.4564	0.7338	0.3699	0.3234	0.5825	0.7448	0.2866
	Video	0.5643	0.4935	0.4890	0.3936	0.6006	0.3615	0.4080	0.4012	0.6079	0.5590	0.6335	0.3436
Oracle	Image	0.3441	0.6830	0.8448	0.6034	0.3296	0.6114	0.8341	0.6143	0.3422	0.6785	0.8155	0.5396
	Video	0.6473	0.5440	0.6275	0.5532	0.6355	0.4460	0.5000	0.5647	0.6510	0.5790	0.6800	0.5248
NaviGen	Image	**0.3346**	**0.6479**	**0.8423**	**0.4960**	0.3173	**0.5879**	**0.8350**	**0.4839**	**0.3390**	**0.6686**	**0.8272**	**0.4180**
	Video	**0.6449***	**0.5343***	**0.5997***	0.4322	**0.6336***	**0.4718***	**0.4468***	**0.4834***	**0.6478***	0.5528	0.6636	**0.4002***



(a) Image Generation



(b) Video Generation

Figure 3: Ablation study on image and video generation.

4.1.3 Implementation Details

All baselines are implemented following their original papers. NaviGen uses Qwen3-1.7B as the base language model. Unless otherwise specified, we use Qwen3.5-Flash for auxiliary method components, including Instruction-Aware Reward, TID, and reasoning generation. For both cold-start embedding initialization and full-parameter finetuning, we set the learning rate to 5×10^{-4}, the warmup ratio to 0.03, and train for 3 epochs, using AdamW optimization with a cosine learning-rate schedule, a maximum sequence length of 2048, and packed SFT examples. For reinforcement learning, we set the group size to 8, the maximum sequence length to 2048, the weight decay to 0.01, the learning rate to 3×10^{-4}, the number of training steps to 600, and the batch size to 480. We employ ViT-B/32 for image encoding and text-embedding-v4 for text encoding. We use GLM-5V-Turbo as the VLM judge, Z-Image-Turbo for 512×512 image generation, and Open-Sora 1.3 for 720p video generation at 24 fps with a total of 81 frames.

4.2 Overall Performance Comparison (RQ1)

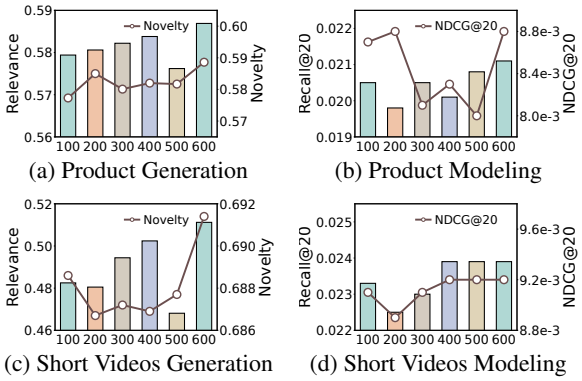
We compare NaviGen against representative baselines on personalized image and video generation; Table 2 reports the results, with Oracle as a non-deployable target-conditioned reference. **Consistent Image-Level Gains.** NaviGen achieves the best image-generation performance on most non-oracle comparisons, leading across all metrics on Games and Short Videos and across Novelty, Aesthetic, and Relevance on Product. The only exception is Product Consistency, suggesting that NaviGen mainly improves personalized target alignment, creative specificity, and visual quality while keeping image-instruction consistency competitive. **Video-Level Transfer and Trade-offs.** NaviGen obtains the best video-generation result in 9 of 12 non-oracle comparisons, indicating that video generation involves a trade-off among frame-level consistency, novelty, visual quality, and preference-specific relevance. **Oracle Reference.** Oracle attains the highest Relevance by directly conditioning on target semantics, yet NaviGen surpasses it on image Aesthetic for Product and Short Videos and nearly matches its video Consistency, suggesting that target semantics alone do not guarantee superior perceptual generation quality. We provide additional human evaluation details in Appendix A.2.

Table 3: Performance comparison on collaborative identifier prediction. Best results are highlighted in bold.

Model	Games				Product				Short Videos			
	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20	R@10	R@20	N@10	N@20
SASRec	0.0281	0.0364	0.0150	0.0171	0.0017	0.0027	0.0006	0.0009	0.0022	0.0039	0.0013	0.0017
TIGER	0.0294	0.0446	0.0150	0.0190	0.0051	0.0109	0.0022	0.0037	0.0025	0.0050	0.0014	0.0021
LC-Rec	0.0463	0.0600	0.0258	0.0289	0.0072	0.0143	0.0040	0.0057	0.0042	0.0080	0.0029	0.0041
OneRec	**0.0471**	0.0605	**0.0267**	0.0288	0.0068	0.0137	0.0033	0.0050	0.0047	0.0111	0.0034	0.0053
NaviGen	0.0431	**0.0621**	0.0247	**0.0294**	**0.0126**	**0.0211**	**0.0066**	**0.0088**	**0.0155**	**0.0239**	**0.0070**	**0.0092**

Table 4: Ablation study on CID prediction modeling.

Variant	Games		Product	
	R@20	N@20	R@20	N@20
w/o TID	0.0546	0.0235	0.0199	0.0083
w/o Init	0.0546	0.0242	0.0191	0.0079
w/o GRPO	0.0613	0.0292	0.0198	0.0081
Init Only	0.0530	0.0233	0.0092	0.0031
w/o Reasoning	0.0604	0.0284	0.0167	0.0066
NaviGen	**0.0621**	**0.0294**	**0.0211**	**0.0088**



(a) Product Generation (b) Product Modeling
(c) Short Videos Generation (d) Short Videos Modeling

Figure 4: Hyperparameter study on GRPO steps.

4.3 Ablation Study (RQ2)

We conduct ablation studies from both personalized generation and CID-space collaborative modeling perspectives, with results shown in Figure 3 and Table 4. **Overall Effect.** The full NaviGen variant achieves the strongest or most balanced results across generation relevance, novelty, and CID modeling metrics, indicating that the proposed modules contribute complementary signals. **Identifier Grounding.** Removing TID grounding or CID initialization weakens both generation and CID modeling performance, while the initialization-only variant performs worst, showing that stable collaborative identifier adaptation must be followed by semantic recovery and full collaborative trajectory modeling. **Reasoning and Reward Alignment.** Removing reasoning supervision hurts CID trajectory modeling and generation quality, and ablating GRPO leads to a weaker relevance–novelty tradeoff and lower CID-space retrieval quality, confirming that transition-level reasoning and multi-task reward alignment jointly improve collaborative-sequence consistency and generation readiness.

4.4 Collaborative Identifier Prediction (RQ3)

We further evaluate NaviGen on **CID2CID** prediction, where the model predicts the next CID from a user's historical CID sequence. This task is not intended to replace TID-based semantic prediction; rather, it isolates whether CID provides an additional view for discovering collaborative signals that are not directly exposed by textual identifiers. Table 3 reports Recall and NDCG on the three datasets. NaviGen achieves the best results on most metrics, with clear advantages on Product and Short Videos, while OneRec remains stronger on the top-10 metrics of Games. Although NaviGen is slightly weaker on Games top-10 retrieval, its advantage at R@20 and N@20 indicates broader top-k coverage in the collaborative identifier space, and its consistent gains on Product and Short Videos demonstrate robustness under different sparsity and item-distribution conditions.

4.5 Hyperparameter Study (RQ4)

We analyze the training-step sensitivity of NaviGen in Table 5 and Figure 4. **SFT Steps.** Increasing supervised tuning mainly improves R@20 across datasets, while consistency only fluctuates within a narrow range, suggesting that longer SFT strengthens CID-space collaborative trajectory modeling without degrading instruction coherence. **GRPO Steps.** RL updates have a clearer effect on generation quality: relevance generally improves toward later steps and novelty remains stable or slightly increases, despite minor mid-training fluctuations. **CID Prediction Stability.** CID collaborative modeling metrics are less sensitive to GRPO, showing mild Product fluctuations and a Short Videos plateau, suggesting that later RL mainly affects generation-side alignment rather than substantially changing CID-space collaborative modeling.

4.6 Case Study (RQ5)

Figure 5 presents a qualitative comparison with TRIPLE on history-guided generation. The user's history evolves from humorous elf interactions to conflict-centered fantasy anime, while the target TID further introduces romance and emotional conflict. TRIPLE captures coarse historical cues such

Table 5: Hyperparameter study on SFT checkpoint steps. Ours corresponds to 7128 steps.

Steps	Games		Product		Short Videos	
	Cons.	R@20	Cons.	R@20	Cons.	R@20
2000	0.3499	0.0554	0.3243	0.0201	**0.3450**	0.0554
4000	0.3429	0.0613	0.3325	0.0198	0.3425	0.0613
6000	0.3502	0.0617	**0.3328**	0.0219	**0.3450**	0.0617
Ours	**0.3505**	**0.0629**	0.3319	**0.0232**	0.3409	**0.0629**

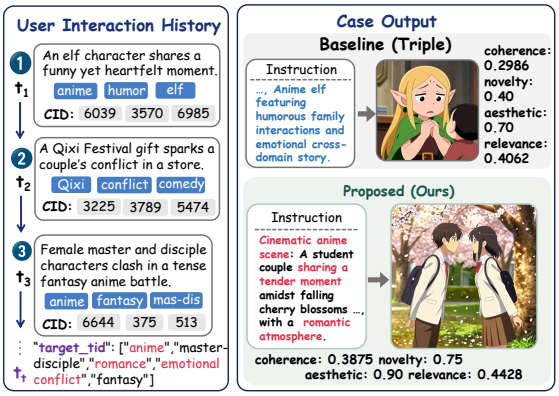


Figure 5: Case study on TRIPLE and our NaviGen

as anime and elf, but its instruction remains generic and fails to reflect the target-side romantic transition. In contrast, NaviGen leverages CID-level collaborative interaction transitions and TID-level semantic grounding to infer the next-interest direction, preserving the anime/fantasy context while specifying visual cues such as a student couple, a tender moment, and a romantic atmosphere. The image achieves higher scores across all metrics, suggesting NaviGen distills collaborative interaction trajectories into generation-ready visual conditions rather than merely copying historical topics.

5 Related Work

Personalized Generation. Personalized generation adapts content to individual users, styles, or contexts, with one line personalizing visual synthesis from histories, retrieved evidence, or recommendation-guided signals (Shen et al., 2024; Xu et al., 2025; Ling et al., 2026). Another line constructs user-specific profiles or alignment signals from edits, demonstrations, or behavioral theories (Gao et al., 2024; Aroca-Ouellette et al., 2025; Noh et al., 2026). While these methods improve user responsiveness, they often rely on profile-like descriptions rather than generation-ready instructions and provide limited coupling between compact behavioral modeling and explicit semantic grounding; in contrast, NaviGen converts implicit interaction histories into executable AIGC instructions for downstream image and video synthesis.

Multimodal Generative Models. Multimodal generative models provide the backbone for turning natural-language instructions into images, videos, and other visual media (Xie et al., 2025; Huang et al., 2025). VLMs increasingly support reasoning and content creation (Hurst et al., 2024; Jin et al., 2025), while AIGC systems continue to improve prompt alignment, visual fidelity, temporal coherence, and motion control (Sun et al., 2025; Seedream et al., 2025; Seedance et al., 2026). However, these models are mainly optimized to follow explicit prompts rather than infer generation conditions from sparse and noisy user behaviors; NaviGen is therefore complementary, translating behavior-grounded preferences into executable instructions for downstream multimodal synthesis.

Behavioral Preference Modeling. Preference modeling learns user interests from behavioral data, with graph and intent-aware models capturing higher-order user–item relations (He et al., 2020; Zhang et al., 2024; Zhao et al., 2025a), and sequential or multimodal methods modeling temporal dynamics and heterogeneous content features (Ren et al., 2024; Fu et al., 2025; Lin et al., 2025). Recent LLM-augmented methods further use language models to interpret item descriptions, refine candidate scoring, simulate users, or calibrate preference decisions (Sun et al., 2023; Qin et al., 2025; Ye et al., 2025). NaviGen instead transforms behavior-derived preference signals into personalized creation by connecting behavior-aware identifier prediction with generation-ready instruction.

6 Conclusion

We presented NaviGen, a behavior-aware framework that turns user interaction history into executable instructions for personalized AIGC. Built on a dual-identifier item representation, NaviGen couples reasoning-augmented supervised tuning with multi-task reinforcement learning to bridge user preference modeling and downstream generation control. By separating CID-based behavioral coding from TID-based semantic grounding, NaviGen treats preference prediction as an intermediate reasoning step rather than an end in itself. Experiments across product, game, and short-video domains demonstrate consistent improvements in personalized image and video generation, instruction quality, and collaborative identifier prediction, establishing a unified path from implicit user behavior to controllable multimodal synthesis.

Limitations

Personalized generation is bounded by how user preference evidence is collected and used in real applications. Interaction histories can contain sensitive behavioral signals, and users may not always expect such signals to be transformed into creative generation conditions. Practical deployments should therefore require explicit opt-in consent, minimize retained histories, anonymize or aggregate logs whenever possible, and provide clear controls for inspecting, editing, or deleting personalized preference profiles. We view personalized instruction generation as an assistive layer for creative control rather than a replacement for deployment-time safety governance.

Ethical Considerations

This work uses existing datasets and item captions for offline research evaluation, without collecting new personal data, inferring sensitive demographic attributes, or deploying personalized generation to real users. The generated instructions are evaluated only as experimental outputs for studying preference-to-instruction modeling. Beyond the privacy and deployment precautions discussed in the limitations, we do not identify additional ethical concerns specific to this work.

References

Ruichuan An, Sihan Yang, Renrui Zhang, Ming Lu, Gaole Dai, Hao Liang, Ziyu Guo, Shilin Yan, Yulin Luo, and Bocheng Zou. 2026. Unictokens: Boosting personalized understanding and generation via unified concept tokens. *Advances in Neural Information Processing Systems*, 38:144638–144664.

Stéphane Aroca-Ouellette, Natalie Mackraz, Barry-John Theobald, and Katherine Metcalf. 2025. Aligning LLMs by predicting preferences from user writing samples. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 1690–1721. PMLR.

Yufan Dang, Chen Qian, Xueheng Luo, Jingru Fan, Zihao Xie, Ruijie Shi, Weize Chen, Cheng Yang, Xiaoyin Che, Ye Tian, and 1 others. 2026. Multi-agent collaboration via evolving orchestration. *Advances in neural information processing systems*, 38:165025–165059.

Jiaxin Deng, Shiyao Wang, Kuo Cai, Lejian Ren, Qigen Hu, Weifeng Ding, Qiang Luo, and Guorui Zhou. 2025. OneRec: Unifying retrieve and rank with generative recommender and iterative preference alignment. *arXiv preprint arXiv:2502.18965*.

Junchen Fu, Xuri Ge, Xin Xin, Alexandros Karatzoglou, Ioannis Arapakis, Kaiwen Zheng, Yongxin Ni, and Joemon M Jose Joemon. 2025. Efficient and effective adaptation of multimodal foundation models in sequential recommendation. *IEEE TKDE*.

Ge Gao, Alexey Taymanov, Eduardo Salinas, Paul Mineiro, and Dipendra Misra. 2024. Aligning llm agents by learning latent preference from user edits. *Advances in neural information processing systems*, 37:136873–136896.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *SIGIR*, pages 639–648.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. Clipscore: A reference-free evaluation metric for image captioning. In *EMNLP*, pages 7514–7528.

Wenxuan Huang, Shuang Chen, Zheyong Xie, Shaosheng Cao, Shixiang Tang, Yufan Shen, Qingyu Yin, Wenbo Hu, Xiaoman Wang, Yuntian Tang, and 1 others. 2025. Interleaving reasoning for better text-to-image generation. *arXiv preprint arXiv:2509.06945*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Zhipeng Jin, Wen Tao, Yafei Li, Yi Yang, Cong Han, Shuanglong Li, and Lin Liu. 2025. Large vison-language foundation model in baidu aigc image advertising. In *KDD*, pages 2303–2312.

Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE.

Yueying Kao, Ran He, and Kaiqi Huang. 2017. Deep aesthetic quality assessment with semantic information. *TIP*, 26(3):1482–1495.

Xixun Lin, Rui Liu, Yanan Cao, Lixin Zou, Qian Li, Yongxuan Wu, Yang Liu, Dawei Yin, and Guandong Xu. 2025. Contrastive modality-disentangled learning for multimodal recommendation. *ACM TOIS*, 43(3):1–31.

Run Ling, Wenji Wang, Yuting Liu, Guibing Guo, Haowei Liu, Jian Lu, Quanwei Zhang, Yexing Xu, Shuo Lu, and Yun Wang. 2026. Ragar: retrieval augmented personalized image generation guided by recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 15278–15286.

Taehyung Noh, Seungwan Jin, Haein Yeo, and Kyungsik Han. 2026. Triple: Theory-driven integration of

planned and habitual behaviors for llm-based personalization. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI-26)*. AAAI Press.

Weicong Qin, Yi Xu, Weijie Yu, Chenglei Shen, Xiao Zhang, Ming He, Jianping Fan, and Jun Xu. 2025. More: A mixture of reflectors framework for large language model-based sequential recommendation. In *Recsys*, pages 299–308.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, and 1 others. 2023. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36:10299–10315.

Xubin Ren, Wei Wei, Lianghao Xia, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024. Representation learning with large language models for recommendation. In *WWW*, pages 3464–3475.

Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, and 1 others. 2026. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*.

Team Seedream, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, and 1 others. 2025. Seedream 4.0: Toward next-generation multimodal image generation. *arXiv preprint arXiv:2509.20427*.

Xiaoteng Shen, Rui Zhang, Xiaoyan Zhao, Jieming Zhu, and Xi Xiao. 2024. Pmg: Personalized multimodal generation with large language models. In *Proceedings of the ACM Web Conference 2024*, pages 3833–3843.

Shangkun Sun, Xiaoyu Liang, Bowen Qu, and Wei Gao. 2025. Content-rich aigc video quality assessment via intricate text alignment and motion-aware consistency. *arXiv preprint arXiv:2502.04076*.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is chatgpt good at search? investigating large language models as re-ranking agents. *arXiv preprint arXiv:2304.09542*.

Saúl Vargas and Pablo Castells. 2011. Rank and relevance in novelty and diversity metrics for recommender systems. In *Recsys*, pages 109–116.

Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. 2025. Show-o: One single transformer to unify multimodal understanding and generation. In *International Conference on Learning Representations*, volume 2025, pages 28240–28264.

Yiyan Xu, Wenjie Wang, Yang Zhang, Biao Tang, Peng Yan, Fuli Feng, and Xiangnan He. 2025. Personalized image generation with large multimodal models. In *Proceedings of the ACM on Web Conference 2025*, pages 264–274.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, and 1 others. 2025. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pages 83048–83077.

Yuyang Ye, Zhi Zheng, Yishan Shen, Tianshu Wang, Hengruo Zhang, Peijun Zhu, Runlong Yu, Kai Zhang, and Hui Xiong. 2025. Harnessing multimodal large language models for multimodal sequential recommendation. In *AAAI*, volume 39, pages 13069–13077.

Yi Zhang, Lei Sang, and Yiwen Zhang. 2024. Exploring the individuality and collectivity of intents behind interactions for graph collaborative filtering. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1253–1262.

Zhiyang Zhang, Junda She, Kuo Cai, Bo Chen, Shiyao Wang, Xinchen Luo, Qiang Luo, Ruiming Tang, Han Li, Kun Gai, and 1 others. 2026. Unleashing the native recommendation potential: Llm-based generative recommendation via structured term identifiers. *arXiv preprint arXiv:2601.06798*.

Chu Zhao, Enneng Yang, Yuliang Liang, Jianzhe Zhao, Guibing Guo, and Xingwei Wang. 2025a. Symmetric graph contrastive learning against noisy views for recommendation. *ACM TOIS*, 43(3):1–28.

Yibo Zhao, Liang Peng, Yang Yang, Zekai Luo, Hengjia Li, Yao Chen, Zheng Yang, Xiaofei He, Wei Zhao, Qinglin Lu, and 1 others. 2025b. Local conditional controlling for text-to-image diffusion models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pages 10492–10500.

Bowen Zheng, Yupeng Hou, Hongyu Lu, Yu Chen, Wayne Xin Zhao, Ming Chen, and Ji-Rong Wen. 2024. Adapting large language models by integrating collaborative semantics for recommendation. In *2024 IEEE 40th International Conference on Data Engineering*, pages 1435–1448. IEEE.

Guorui Zhou, Honghui Bao, Jiaming Huang, Jiaxin Deng, Jinghao Zhang, Junda She, Kuo Cai, Lejian Ren, Lu Ren, Qiang Luo, and 1 others. 2025. Openonerec technical report. *arXiv preprint arXiv:2512.24762*.

A Appendix

A.1 Baseline Methods

To ensure a comprehensive study, we compare NaviGen against a broad set of baselines covering personalized generation, collaborative filtering and broader behavioral preference modeling, and prompting-based references.

Personalized Generation Methods

- **PMG** (Shen et al., 2024): LLM-extracted user cues condition multimodal generators for personalized content synthesis, guided by behavior-aware prompts.
- **PIGEON** (Xu et al., 2025): Retrieved preference evidence steers frozen generation agents without model fine-tuning, preserving efficient deployment workflows.
- **RAGAR** (Ling et al., 2026): Semantic retrieval weights relevant histories, while ranking feedback balances personalization and fidelity across interaction-rich scenarios.
- **CIPHER** (Gao et al., 2024): Historical user edits are retrieved to infer preferences and align generated outputs with user intent through edit-aware context matching.
- **PROSE** (Aroca-Ouellette et al., 2025): Iterative refinement and consistency checks infer user preferences from demonstrations, using structured self-verification loops.
- **TRIPLE** (Noh et al., 2026): A theory-driven LLM-based personalization framework integrating planned and habitual behavior modeling for preference reasoning.

Behavioral Preference Modeling Methods

- **SASRec** (Kang and McAuley, 2018): Transformer-based sequential preference model learning user behavior from item ID sequences via self-attention.
- **TIGER** (Rajput et al., 2023): A generative item-ID modeling framework that converts item IDs into tokens and models sequential preference prediction as next-token generation.
- **LC-Rec** (Zheng et al., 2024): An ID-based sequential preference modeling method that enhances representation learning with additional latent/contextual signals.

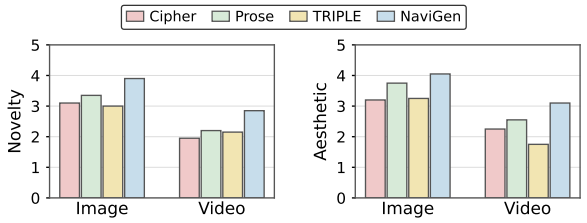


Figure 7: Human evaluation on created content.

- **OpenOneRec** (Zhou et al., 2025): It integrates item-text alignment into an end-to-end generative preference modeling framework for scalable preference prediction and reasoning.

Prompting and Reference Baselines

- **NPC** : No-preference conditioning removes user evidence, such as reference images and similar historical items, using a generic prompt for generation to isolate personalization effects.
- **Oracle** : Ground-truth target semantics, such as the target caption or TID, are used as a non-deployable upper-bound reference for diagnostic comparison only.

A.2 Human Evaluation on Content

To complement automatic VLM-based evaluation, we further conduct a human study to directly assess the perceptual quality of NaviGen. We perform human evaluation which averaged over three domains, including Product, Short Videos, and Games. From each dataset, we randomly sample 20 image cases and 5 video cases for evaluation. For NaviGen and three representative generation baselines, CIPHER, PROSE, and TRIPLE, which all support both image and video generation, we recruit 24 student volunteers to evaluate anonymized and randomly shuffled outputs using a 5-point Likert scale along two dimensions: Novelty, reflecting the creativity and non-triviality of the visual interpretation, and Aesthetic quality, capturing visual appeal, composition, clarity, and overall polish. Scores from 1 to 5 indicate very poor, poor, fair, good, and excellent quality, respectively. The evaluation process takes approximately 1.5 hours to complete. As shown in Figure 7, NaviGen consistently achieves higher average ratings on both dimensions in image and video settings, suggesting that behavior-conditioned instructions lead to more creative and visually appealing personalized content.

(a) Image: Game

(b) Image: Product

(c) Video: Short Video

Figure 6: Comprehensive generation cases across three representative domains, each with its target tid.

A.3 Prompt Templates

We apply these prompts to six main works and two reasoning cases (shown at last), including:

- **TID Generation:** Fig. 9 converts item captions into compact semantic TIDs.
- **Thinking Generation:** Fig. 14 generates reasoning from historical TIDs to the target TID.
- **Evolutionary Search:** Fig. 8 searches for the best target-aligned AIGC prompt.
- **Oneshot Distillation:** Fig. 10 distills history and so on into final first-person reasoning.
- **SFT Task Prompts:** Fig. 11 defines SFT tasks for ID mapping, next-item prediction, and AIGC instruction generation.
- **GRPO/RL Task Prompts:** Fig. 12 defines preference prediction and instruction generation.
- **CID2CID/CID2INS Reasoning:** Fig. 13 shows our reasoning process.
- **Novelty/Aesthetic Judging:** Fig. 15 illustrates the evaluation of novelty and aesthetics for image (a) and video (b) generation.

Evolutionary Search

SYSTEM_PROMPT = """You are an AIGC creative search teacher model.
Your task is to search for the best AIGC creative instruction for the target item.

Search principles:
1. The target item is the primary objective. Historical TIDs are only auxiliary clues.
2. Each candidate must implicitly point to the target item, but the creative instruction must not explicitly mention, copy, or quote the target item text or tokens.
3. Each candidate must also output a concise reasoning explaining why the prompt is suitable and what information it used.
4. Use multiple rounds of selection, crossover, and mutation to refine the candidate set.
5. Output only one JSON object with fields "prompt" and "reasoning"."""

SCORER_SYSTEM_PROMPT = """You are a strict AIGC prompt evaluation model.
Given the user's historical TIDs, the target TID, and a candidate prompt, score the candidate independently on the following four dimensions:
1. consistency: whether it is centered on the target item semantics, uses history only as supporting context, and avoids drifting or explicitly naming the target.
2. novelty: whether it introduces reasonable innovation without losing the intended direction.
3. aesthetic: whether it shows clear and layered visual composition ability, including subject, scene, style, camera language, lighting, mood, and key details.
4. executability: whether it is specific, clear, contradiction-free, and directly usable for AIGC generation.

Scoring requirements:
1. Output a floating-point score from 0 to 10 for each dimension.
2. The four dimensions must be judged independently; do not give all high scores just because you like the candidate overall.
3. The reasoning should briefly explain the main strengths and weaknesses and stay within 200 characters.
4. If the prompt explicitly mentions the target item, lower consistency and executability.
5. Output only one JSON object. Do not output Markdown and do not output any extra explanation.

Output format:
{ "consistency": 0.0, "novelty": 0.0, "aesthetic": 0.0, "executability": 0.0, "reasoning": "..." }"""

Figure 8: Generation prompts create target-aligned AIGC candidates, while scoring prompts evaluate and select the best final prompt.

Tid Generation

SYSTEM_PROMPT = f"""You are a cross-domain item identifier generator.
Your task is to compress the input caption into a Term ID (TID): a set of structured, standardized, semantically dense terms.

Goals:
1. The terms should cover the core semantics of the item.
2. Preserve key information while staying concise and avoiding redundant modifiers.
3. Prefer stable cross-domain signals such as:
   - subject or category - core scene or use case
   - key attributes or selling points
   - style, atmosphere, or audience only when truly important
4. Use standard, general, reusable wording. Avoid synonym stacking and avoid sentence-style expressions.
5. You decide how many terms are needed, but the result must be expressive enough and must not exceed {MAX_TERMS} terms.
6. Each term may be a single word or a very short phrase, but the overall result must stay compact.
7. Order the terms by importance, with the most essential term first.

Strict output requirements:
1. Output JSON only.
2. The format must be {{"terms": ["term1", "term2"]}}.
3. Do not output explanations, reasoning process, Markdown, or extra text.
4. All terms must be in English.
5. Do not use Chinese or any other non-English language in the terms."""

Figure 9: Convert item captions into structured Term IDs.

Oneshot Distillation

SYSTEM_PROMPT = """You are an AIGC creative planning teacher model.
Your task is to distill a final reasoning paragraph from the user's historical TIDs, the step1 reasoning, the target TID, and the step2 evolution summary.

Strict requirements:
1. The output must be one coherent paragraph of natural-language reasoning written as first-person internal thinking, with "I" as the subject.
2. I should first infer, from the history and the step1 reasoning, what kind of next item the user is likely to engage with, without naming the final target too early.
3. Then I should summarize how the prompt-writing route gradually moved closer to the right answer by comparing early drafts, keeping stronger parts, rewriting weaker parts, and refining the wording.
4. Do not use biological evolution terms such as selection, crossover, mutation, founder, or similar jargon. Use plain language instead.
5. Near the end of the paragraph, I may explicitly mention the final target TID and explain why the route converges there.
6. The paragraph should explain how the inferred target direction, the history clues, and the prompt revision route together support the final prompt-writing idea.
7. Do not output XML/HTML tags, Markdown code blocks, bullet points, titles, or template fields.
8. Do not write as a teacher note, analysis report, or explanation addressed to someone else. The paragraph must read like the student model's own think content."""

USER_PROMPT_TEMPLATE = """User historical TIDs:
{history_tids}

Step1 reasoning from history to the next target:
{step1_reasoning}

Target TID:
{target_tid}

Step2 prompt revision summary:
{evolution_summary}

Based on the information above, output one final reasoning paragraph.

Requirements:
1. Write in first person, using "I" as the subject throughout.
2. You may condense the step1 reasoning, but keep its core judgment about the likely next-item direction.
3. Use the step2 summary to explain in plain language how the drafts moved closer to the right final prompt.
4. Do not use technical or biological evolution wording such as selection, crossover, mutation, founder, elite, mate, or similar labels.
5. Near the end, explicitly mention the target TID and explain why the route finally points there.
6. End by summarizing how I would write the final prompt around that target direction, using history as support rather than as the main objective.
7. Output one complete paragraph only."""

Figure 10: Distills user history, target reasoning, and prompt refinement into final first-person reasoning.

SFT Task

"cid2tid": (
"You are an ID mapping assistant. "
"Your task is to map the input cid to the corresponding tid (tid is the item's metadata). "
"The final answer must be a JSON object only, with the field target_tid and no other fields."
),

"tid2cid": (
"You are an ID mapping assistant. "
"Your task is to map the input tid (tid is the item's metadata) to the corresponding cid. "
"The final answer must be a JSON object only, with the field target_cid and no other fields."
),

"cid2cid": (
"You are a personalized recommendation assistant. "
"Predict the target cid interaction based on user history cid interaction. "
"The final answer must be a JSON object only, with the field target_cid and no other fields."
),

"cid2ins": (
"You are a personalized recommendation and AIGC instruction generation assistant. "
"Based on user history cid interaction, output user target tid interaction and the AIGC instruction . "
"Think step by step, then answer. The final answer must be a JSON object only, with the fields target_tid and target_ins."
),

Figure 11: Four prompts of SFT tasks' construction.

GRPO Task

"cid2cid": (
"You are a personalized recommendation assistant. "
"Predict the target cid interaction based on user history cid interaction. "
"Think step by step, then answer. "
"The final answer must be a JSON object only, with the field target_cid and no other fields."
),

"cid2ins": (
"You are a personalized recommendation and AIGC instruction generation assistant. "
"Based on user history cid interaction, output user target tid interaction and the AIGC instruction. "
"Think step by step, then answer. "
"The final answer must be a JSON object only, with the fields target_tid and target_ins."
),

user = (
"Task: Predict the target_cid / Generate the target recommendation result based on the user's historical interactions in hist_cid.\n"
f"hist_cid: {_format_hist_cid(hist_cid)}\n"
"Think first, then output JSON only."
)

Figure 12: Two prompts of GRPO tasks' construction.

Reasoning cases

"cid2sid": I start by analyzing the user's history, which is heavily dominated by video game titles and specific console ecosystems like PlayStation 4 and Xbox, alongside family-friendly franchises such as...Therefore, based on this progression from software consumption to necessary hardware maintenance for the identified console ecosystem, I conclude with the target_tid: Xbox One, Controller Charging Station, Dual Charger, Rechargeable Battery Packs, PDP Gaming, Energizer, Gaming Accessory.

"cid2ins": By reviewing my past interactions, I notice a consistent thread of interest in female subjects within romantic or social settings, ranging from...where a couple engages in a lively, humorous exchange, using dynamic lighting and handheld camera movements to evoke the festive yet intimate spirit of the occasion, ensuring that the historical preference for romance and lifestyle content serves as a supportive backdrop rather than the sole focus.

Figure 13: Cases of reasoning

Thinking Generation

SYSTEM_PROMPT = """You are a recommendation reasoning teacher model.
Your task is to write a student-model-style reasoning paragraph that starts from the user's historical TIDs and finally points to the given target_tid.

Strict requirements:
1. The output must be one complete, natural paragraph of reasoning text.
2. The reasoning must be written from the perspective of a student model (use first-person thinking style such as "I should...", "I need to...").
3. The first part of the paragraph should explain how to extract semantic clues from history and narrow down candidates toward the correct target_tid.
4. The final part of the paragraph should explicitly conclude with the provided target_tid as the final decision.
5. Do not output any extra explanation, list, or metadata; output only the single reasoning paragraph."""

Figure 14: Reason from hist TIDs to target TID.

Judging: Image

You are AIM-judge for AIGC image generation evaluation.
Evaluate the generated media conditioned on the creative text instruction (ins).
Return JSON only with fields: "novelty_score", "aesthetic_score", "reasoning".
Use a 0-1 scale.
NOVELTY (0-1): How creatively does the image interpret the instruction?
A concise instruction leaves more room for creative interpretation — reward images that make clever, inventive choices beyond the literal words.
Do NOT equate detail-count in the instruction with novelty; a short but imaginative prompt that yields a distinctive image deserves a high score.
Reward visual creativity, unexpected yet fitting compositions, and artistic interpretation.

AESTHETIC (0-1): Evaluate purely on visual appeal — clarity, color harmony, composition, texture, lighting, and overall polish.
The subject matter (product vs. landscape) should NOT bias your score; a well-executed product shot deserves the same score as a dramatic scene of equal technical quality.

Media type: {media_type}.
Instruction (ins): {ins}

(a) Image generation evaluation

Judging: Video

You are AIM-judge for AIGC video generation evaluation.
Evaluate the generated video conditioned on the creative text instruction (ins).
The attached images are evenly sampled frames from the generated video, shown in chronological order.
Return JSON only with fields: "novelty_score", "aesthetic_score", "reasoning".
Use a 0-1 scale.

NOVELTY (0-1): How creatively does the video interpret the instruction?
A concise instruction leaves more room for creative interpretation — reward videos that make clever, inventive choices beyond the literal words.
Do NOT equate detail-count in the instruction with novelty; a short but imaginative prompt that yields a distinctive video deserves a high score.
Reward visual creativity, unexpected yet fitting compositions, artistic interpretation, and coherent use of multiple frames over time.

AESTHETIC (0-1): Evaluate purely on visual appeal across the sampled frames — clarity, color harmony, composition, texture, lighting, temporal consistency, motion/camera coherence, and overall polish.
The subject matter (product vs. landscape) should NOT bias your score; a well-executed product shot deserves the same score as a dramatic scene of equal technical quality.

Media type: video. The images are sampled frames and should be judged together as one video, not as unrelated images.
Instruction (ins): {ins[:1600]}

(b) Video generation evaluation

Figure 15: AIM-judge prompts for novelty and aesthetic evaluation. Both static images (a) and dynamic video sequences (b) are appraised for their creative interpretation of instructions (novelty) and visual polish (aesthetics).