

HASTE: Hierarchical and Attribute-aware Supervision for Generative Recommendation

Tripti Shukla
tripti.shukla@glance.com
Glance
Bengaluru, India

Shwetha Somasundaram
shwetha.s@glance.com
Glance
Bengaluru, India

Akshat Gupta
akshat.gupta@glance.com
Glance
Bengaluru, India

Rohit Garg
rohit.garg@glance.com
Glance
Bengaluru, India

Paul Duff
paul.duff@glance.com
Glance
Vilnius, Lithuania

Ian Anderson
ian.anderson@glance.com
Glance
London, United Kingdom

Abstract

Generative recommendation models have shown promising results by representing items as semantic IDs and generating these IDs autoregressively. We introduce an approach showing that explicitly incorporating hierarchical and attribute-level supervision into generative recommendation can simultaneously improve training efficiency and long-tail recommendation performance. Existing approaches typically optimize semantic ID prediction with a standard causal cross-entropy objective, which does not explicitly exploit the hierarchical structure of semantic IDs and can require substantial training to learn effective item representations. Moreover, data-sparse tail items often suffer from poor representations due to limited supervision, resulting in weaker long-tail recommendation performance. We address these challenges with HASTE (Hierarchical and Attribute-aware Supervision for generative Recommendation), a unified training framework that introduces hierarchical and attribute-specific supervision. HASTE’s Hierarchical Loss explicitly exploits the multi-level structure of semantic IDs, providing supervision at each level of the item hierarchy and substantially accelerating training convergence. Its Attribute-Specific Losses encourage the model to preserve fine-grained item attributes, leveraging the language model’s inherent semantic knowledge to improve representations and recommendations for tail items. We extensively evaluate HASTE on a fashion recommendation benchmark and across multiple domains of the Amazon dataset. HASTE improves recommendation performance across datasets and item-frequency regimes. Notably, the Hierarchical Loss achieves approximately $2\times$ faster convergence than the standard causal cross-entropy objective, substantially improving training efficiency while maintaining strong recommendation quality.

CCS Concepts

• **Information systems** → **Recommender systems**; Personalization; • **Computing methodologies** → *Machine learning*; *Deep learning*.

Keywords

Generative Recommendation, Semantic IDs, Hierarchical Supervision, Attribute-aware Recommendation, Long-tail Recommendation

ACM Reference Format:

Tripti Shukla, Shwetha Somasundaram, Akshat Gupta, Rohit Garg, Paul Duff, and Ian Anderson. 2026. HASTE: Hierarchical and Attribute-aware Supervision for Generative Recommendation. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 16 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 Introduction

Large language models (LLMs) have triggered a paradigm shift across modalities, replacing task-specific pipelines with a single model that is pre-trained once and then adapted through instructions or lightweight fine-tuning [3, 33]. Recommender systems are undergoing an analogous shift. Industrial recommenders are conventionally built as cascades, commonly described as matching, pre-ranking, ranking, and re-ranking [23, 27], whose stages are developed and trained as separate models with their own objectives: YouTube’s two-stage system, for instance, trains candidate generation as an extreme multiclass classification problem and ranking under a watch-time-weighted logistic objective [5], while pre-ranking [37] and re-ranking [25] each introduce objectives of their own. Generative recommendation instead casts the entire retrieval problem as a single sequence-to-sequence task: given a user’s interaction history, the model *generates* the identifier of the next item directly. [10] TIGER [26] established the form this now takes, replacing arbitrary item IDs with residual-quantized semantic IDs. Subsequent work has shown that ranking and retrieval can be unified in a single generative backbone [7, 40], whose quality scales as a power law of training compute [40].

A parallel line of work marries generative recommendation with LLM backbones in order to exploit the world knowledge and semantic priors that recommendation data alone cannot supply [11, 41]. The bridge between the two is the *semantic ID* (SID): rather than

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference acronym 'XX, Woodstock, NY

© 2026 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2018/06
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

assigning each item an atomic, randomly initialised embedding, an item is represented as a short sequence of discrete codes derived from its content representation. Semantically similar items share prefixes, so the identifier space itself carries item semantics, and the model can generalise to items with little or no interaction history [26]. Semantic IDs are typically produced by residual quantization—either RQ-VAE [17, 26] or the training-free residual k -means of Quantitative Alignment Multi-Modal Recommendation [21]—and are therefore *hierarchical by construction*. Crucially, this makes the levels *asymmetric*: because every later code quantizes the residual left by the earlier ones, an error at a coarse level redefines the target of every level below it, whereas an error at the finest level perturbs only a local refinement.

Despite this structure, essentially all existing generative recommenders supervise semantic-ID generation with a flat causal cross-entropy (CE) loss over the extended vocabulary. The consequence is not merely that the model must *rediscover* the coarse-to-fine organisation residual quantization already encoded—it is that flat CE loss actively *erodes* it. We show in Section 4 that fine-tuning under flat CE cuts the cosine coupling between adjacent SID levels by 35% relative to initialization (from 0.55 to 0.36), while a level-aware objective retains it at 0.50, a 9% reduction, and induces 2–4× less embedding drift from the shared initialization.

A second, orthogonal limitation is that the supervision an item receives is proportional to how often it appears as a training target. Head items accumulate enough gradient to acquire useful representations; data-sparse tail items do not. This matters because semantic IDs are quantized from *holistic* item embeddings, so they organise the catalogue by overall similarity rather than by any single attribute: a property a user actually asks for (color, material, fit, pattern) is smeared across levels rather than localised. Measuring colour directly, the coarsest code alone yields only 37% mean colour purity while conditioning on the first two codes raises it to 83%, and a probing analysis shows the information is present but not linearly organised—a linear probe underperforms the catalogue-mean baseline at every SID position, whereas a small MLP probe reaches $\Delta E = 38.5$ against a baseline of 56.3. The signal is latent in the representation and unexploited by the generation objective. Yet the LLM backbone already *knows* what a “ribbed burgundy wool cardigan” is: the interaction log is the bottleneck, not the backbone’s semantic knowledge.

To address these limitations, we propose **HASTE** (Hierarchical and Attribute-aware Supervision for generative Recommendation), a fine-tuning framework that introduce two complementary objectives. The **Hierarchical Loss** replaces flat causal CE with level-aware supervision: coarser levels carry larger base weights, and each level’s weight is gated by the model’s *detached* confidence in the upstream codes it is conditioned on, so fine-grained supervision is amplified only once the coarse decisions it depends on are actually resolved. Because confidence estimates are unreliable early in training, the objective anneals from purely static hierarchical weighting to fully confidence-gated weighting. Scheduled sampling [1] further perturbs the upstream prefix, exposing the model to the upstream errors it is likely to encounter during decoding rather than only to clean, teacher-forced trajectories. The **Attribute-Specific Losses** attach lightweight MLP regression heads to the hidden state already used to predict the fine-grained

SID code, and supervise them against catalogue-derived attribute targets. A frequency-aware weighting further concentrates the signal on infrequent items. We instantiate the framework for two complementary attributes—perceptual (CIELAB colour [4]) and compositional (material composition)—and find that the gains land where interaction supervision is weakest: mean top-1 colour error drops from 54.5 to 50.2 on tail colours and 33.5 to 29.0 on torso colours, while head colours are unchanged (12.5 to 12.6).

Our contributions are as follows:

- We propose a **Hierarchical Loss** that supervises semantic-ID generation level-by-level, combining decreasing level weights, detached confidence gating on upstream codes, an annealed transition from static to conditional weighting, and scheduled sampling of the upstream prefix to close the train–inference exposure gap. It exploits the hierarchy explicitly rather than leaving it to be rediscovered, preserves the inherited structure, and yields $\sim 2\times$ faster convergence than causal CE at equal or better recommendation quality.
- We propose **Attribute-Specific Losses** that inject fine-grained item-attribute supervision into the generative recommender through auxiliary heads on the fine-grained SID prediction state. (Section 5.2).
- We evaluate HASTE on three Amazon Review [13] subsets under the standard P5 5-core protocol [10], on Fashion-140k, and on an internal production-scale fashion catalog, against sequential, generative, and LLM-based baselines. HASTE gives prominent gains across datasets and item-frequency regimes, with the largest improvements on the tail items.

2 Related Works

Sequential and Generative Recommendation. Sequential recommenders predict the next item from an ordered interaction history. GRU4Rec [12] models sessions with recurrence, while SAS-Rec [15] and BERT4Rec [29] use unidirectional and bidirectional self-attention respectively. All three represent an item as an atomic embedding in a table that grows with the catalogue, so a rare item’s representation is learned almost entirely from its few occurrences, and retrieval needs a nearest-neighbour index maintained separately from the model. Generative retrieval removes the index: the model decodes the target’s identifier token by token [31]. GENRE [6] first cast retrieval this way, generating entity names autoregressively; DSI [32] replaced names with structured semantic identifiers obtained by hierarchical clustering; and NCI [35] added a prefix-aware decoder. P5 [10] first brought this seq2seq formulation to recommendation, and TIGER [26] specialised it to semantic IDs, quantizing content embeddings with RQ-VAE [17] and decoding the next item’s codes from the history; because those codes are derived from content and shared prefix-wise, a rare item is no longer represented in isolation. HSTU [40] showed the paradigm scales as a power law of training compute, and OneRec [7] and OneRec-Think [20] deploy it at industrial scale. In addition to the generation framework, item tokenization has also attracted considerable attention. Recforest [8] applies hierarchical k -means to item text embeddings and uses the resulting cluster path as the identifier, while QARM [21] replaces the learned quantizer with a training-free

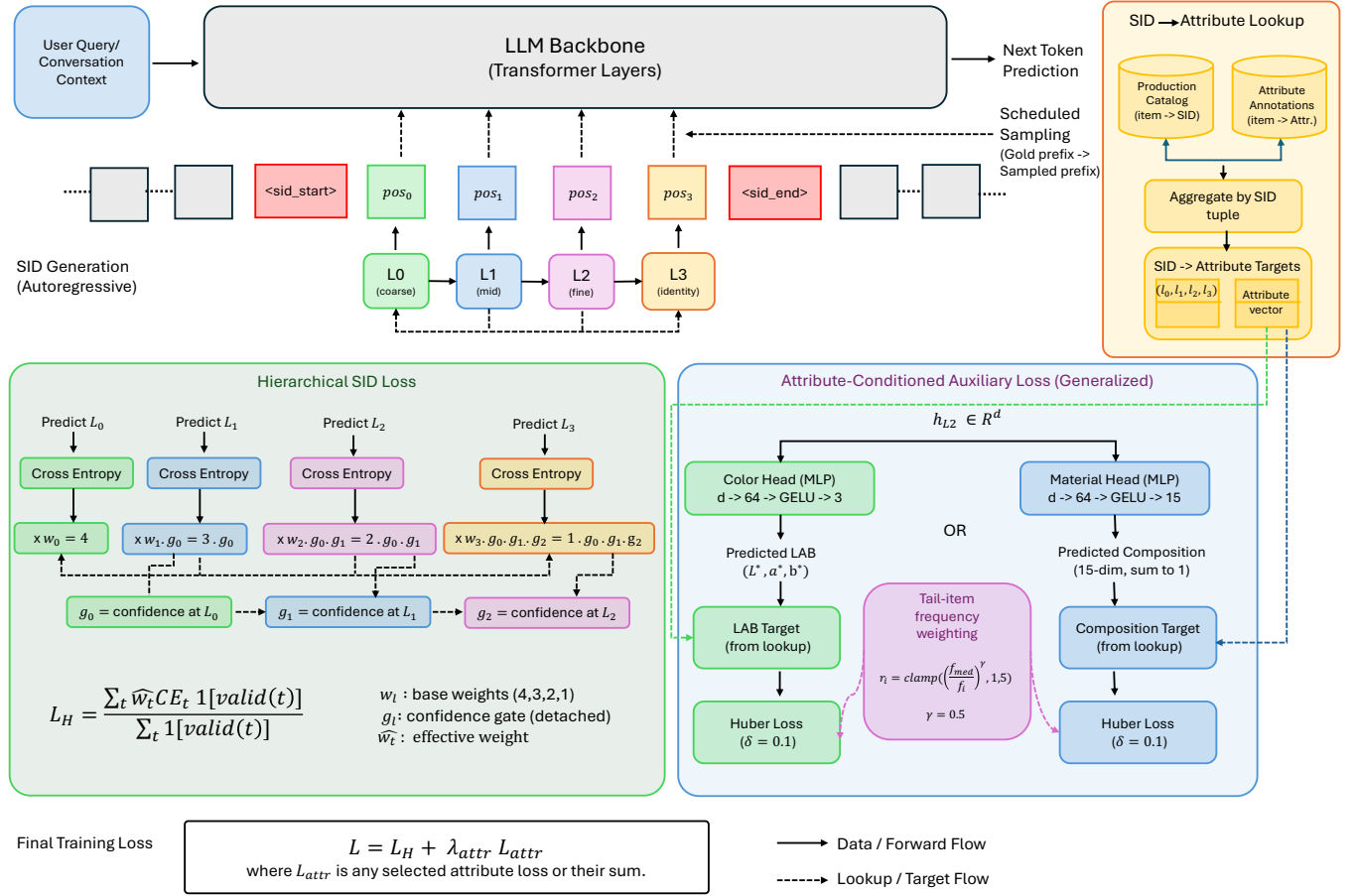


Figure 1: Overview of the proposed HASTE training framework. The LLM autoregressively predicts hierarchical SIDs L_0, L_1, L_2, L_3 with confidence-gated loss L_H , where w_i and g_i denote level weights and detached confidence gates. The LLM hidden state $h_{L2} \in \mathbb{R}^d$ feeds a selected attribute-specific MLP, with L_{attr} weighted by λ_{attr} . The SID tuple (l_0, l_1, l_2, l_3) indexes its corresponding attribute vector.

residual k -means. Furthermore, several studies incorporate collaborative signals into the tokenization process: MMGR [19] performs quantization over a user–item interaction graph, EAGER [36] decodes behavioural and semantic tokens in two parallel streams, and LETTER [34] augments a learnable tokenizer with collaborative and diversity regularisers.

LLMs for Recommendation. Chat-Rec [9] prompts a general-purpose model with textual interaction histories, and GPT4Rec [18] fine-tunes one to generate item titles; such models rank competently only over small candidate pools and remain sensitive to interaction order, item position, and popularity [14]. Recommenders built on language-model backbones tune on recommendation data instead: P5 [10] unifies several tasks as text-to-text but with arbitrary numeric item IDs, LC-Rec [41] aligns semantic-ID tokens with language through auxiliary alignment tasks, and IDGenRec [30] generates textual identifiers instead of quantized codes. This work follows the semantic-ID line and investigates the *training objective* itself, given a fixed architecture. Whereas LC-Rec introduces

additional alignment *tasks*, HASTE reshapes the loss on the recommendation task and derives attribute targets from the catalogue, requiring no additional task data or prompt engineering.

Structured and Auxiliary Supervision. Label structure is well exploited outside recommendation. Hierarchical softmax [24] factorises the prediction over a tree, coarse-to-fine curricula [28] order the levels in time, and hierarchy-aware penalties [2] charge more for distant errors than for sibling errors. Auxiliary objectives likewise densify supervision: deeply-supervised nets [16] attach classifiers at intermediate layers, MMoE [22] shares experts across tasks in the multi-task architectures common to recommenders, and CLCRec [38] uses a contrastive objective to preserve collaborative signal in the content representations of cold-start items. Scheduled sampling [1] addresses the exposure bias of autoregressive decoders, and applies to semantic IDs, whose levels are defined relative to their predecessors. Semantic IDs are a residual hierarchy over an autoregressively decoded sequence, so all three ideas apply directly. HASTE combines level-conditioned supervision over SID

codes with attribute heads that recycle the backbone’s semantic knowledge into tail-item representations.

3 Preliminaries

3.1 Generative Recommendation with Semantic IDs

Generative recommendation [26] formulates item retrieval as a sequence generation problem, where a language model generates the identifier of a recommended item conditioned on the user’s interaction history. Recent approaches represent items using *semantic IDs* (SIDs), replacing independent item identifiers with sequences of discrete tokens derived from the item’s semantic representation. For an item i , its semantic ID is represented as

$$\text{SID}_i = \langle \text{sid_start} \rangle s_{i,0} s_{i,1} s_{i,2} s_{i,3} \langle \text{sid_end} \rangle, \quad (1)$$

where $s_{i,0}, \dots, s_{i,3}$ denote the semantic-ID codes and the start and end tokens delimit the SID sequence.

Given a user interaction context x , the language model autoregressively generates the semantic-ID codes. The probability of generating the SID can be factorized as

$$P(\text{SID}_i | x) = \prod_{l=0}^3 P_\theta(s_{i,l} | x, s_{i,<l}), \quad (2)$$

where $s_{i,<l}$ denotes the semantic-ID codes generated before level l . The generated SID is subsequently mapped to the corresponding item in the catalog. Unlike conventional item-ID-based recommendation, semantic IDs allow the model to operate over a structured representation of the item space. The effectiveness of this formulation therefore depends not only on the semantic structure encoded by the SIDs, but also on how effectively the training objective exploits this structure during generation.

3.2 Residual Quantization for Semantic IDs

Semantic IDs can be obtained using residual quantization [39], which represents an item embedding through a sequence of progressively refined discrete codes. Given an item representation z , residual quantization first selects a code from a coarse codebook and subsequently quantizes the residual using additional codebooks. The resulting representation can be expressed as

$$z \approx e_{s_{i,0}}^{(0)} + e_{s_{i,1}}^{(1)} + e_{s_{i,2}}^{(2)} + e_{s_{i,3}}^{(3)}, \quad (3)$$

where $e_{s_{i,l}}^{(l)}$ denotes the code selected from the l -th quantization codebook.¹

This construction induces a natural coarse-to-fine hierarchy among the semantic-ID codes: because $s_{i,l}$ is chosen to quantize the residual $z - \sum_{k<l} e_{s_{i,k}}^{(k)}$ left after the first l levels, level l ’s code is defined only relative to the codes chosen at levels $0, \dots, l-1$. Consequently, earlier codes capture broader semantic characteristics of the item, while subsequent codes encode residual information that progressively refines the representation, and the semantic meaning of a later code is conditioned on the codes that precede it. A key implication, which we return to in Section 3.3, is that an incorrect

code at an early level changes the residual every later level is quantizing against — so errors at coarse levels are structurally more consequential than errors at fine levels.

3.3 Semantic-ID Prediction with Causal Language Models.

Under the standard next-token prediction objective, each semantic-ID code is supervised using causal cross-entropy, summed over the training corpus $\mathcal{D}$:

$$\mathcal{L}_{\text{CE}} = - \sum_{i \in \mathcal{D}} \sum_{l=0}^3 \log P_\theta(s_{i,l} | x, s_{i,<l}). \quad (4)$$

The objective provides supervision for each semantic-ID prediction while preserving the autoregressive generation process. However, it weights the loss at every level equally: a mistake on the coarse code $s_{i,0}$ and a mistake on the fine code $s_{i,3}$ contribute the same magnitude to $\mathcal{L}_{\text{CE}}$. This uniform weighting is at odds with the coarse-to-fine structure established in Section 3.2 since an error at an early level corrupts the semantic context for every level after it, while an error at a late level only affects that level’s refinement, the two errors are not equally costly.

This distinction is important in semantic-ID-based recommendation: the training objective does not reflect the hierarchical relationship induced by residual quantization, even though the model’s inputs and outputs do. We build on this observation by introducing hierarchy-aware supervision in HASTE.

4 Motivation

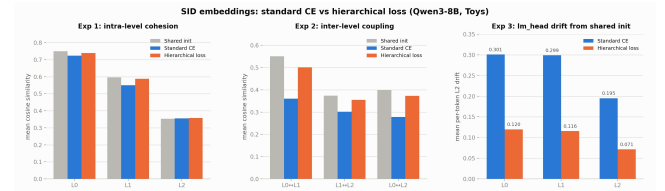


Figure 2: Hierarchical structure of semantic-ID representations under different fine-tuning objectives. (a) Within-level cohesion, (b) inter-level coupling, and (c) embedding drift from the shared initialization. Hierarchical Loss better preserves the structure inherited from residual quantization than standard causal cross-entropy.

Preserving the Hierarchical Structure of Semantic IDs. Semantic IDs (SIDs) generated by residual quantization encode items through a coarse-to-fine hierarchy, where earlier codes capture broad semantics and later codes progressively refine the representation. However, standard causal cross-entropy treats SID tokens as independent vocabulary predictions and does not explicitly model dependencies across hierarchy levels. This raises the question of whether the structure encoded by RQ-VAE is preserved during language-model fine-tuning.

We compare models trained with standard causal cross-entropy and our Hierarchical Loss, both initialized from the same vocabulary-aligned checkpoint, obtained after extending the language model

¹We treat all codebooks as comparably sized and expressive; this is the common setup in RQ-VAE-based SID construction and is not a requirement of residual quantization itself.

vocabulary with the SID tokens and training their embeddings to align with the pretrained representation space, and subsequently subjected to full-parameter supervised fine-tuning. As shown in Figure 2, we analyze the SID representations along three complementary dimensions: within-level cohesion, inter-level coupling, and embedding drift from the shared initialization. Standard CE substantially weakens the hierarchical structure: the average cosine similarity between L0 and L1 output embeddings decreases from 0.55 at initialization to 0.36 after fine-tuning, a 35% reduction. In contrast, Hierarchical Loss preserves this relationship at 0.50, corresponding to only a 9% reduction. We observe similar trends across other level relationships, while the coarse L0 representations also retain greater within-level cohesion under hierarchical supervision.

Moreover, relative to the shared initialization, Hierarchical Loss induces 2–4× less embedding drift than standard CE across all SID levels and embedding tables. We measure drift as the mean per-token Euclidean distance $\|v_{\text{finetuned}} - v_{\text{init}}\|_2$ between each SID token’s row vector after fine-tuning and its value at the shared initialization. These results suggest that a flat token-level objective can weaken the coarse-to-fine structure inherited from residual quantization. This motivates our **Hierarchical Loss**, which explicitly incorporates dependencies across SID levels, encouraging the model to learn coarse semantics before progressively refining them into fine-grained item representations.

Improving Tail-Item Representations with Attribute Supervision. While hierarchical supervision addresses the structure of semantic IDs, it does not explicitly ensure that the learned representations preserve individual item attributes. This is particularly important for generative recommendation, where users may express preferences for specific attributes such as color, material, fit, or pattern. Semantic IDs are derived from holistic item representations and therefore organize products according to overall semantic similarity rather than any single attribute. Consequently, an attribute relevant to a user’s query may be distributed across multiple levels of the semantic-ID hierarchy and need not be explicitly preserved by the standard next-token prediction objective.

hierarchy: the L0 code alone achieves only 37% mean color purity, whereas conditioning on the (L0, L1) prefix increases color purity to 83%. This indicates that fine-grained color information emerges progressively across the hierarchy rather than being localized to a single SID level. Moreover, a probe analysis shows that color information is present in the model’s hidden representations but is not readily accessible through a simple linear mapping. While the linear probe performs worse than the catalog-mean baseline at all SID positions, a lightweight MLP substantially improves color prediction, achieving a ΔE of 38.5 at L2 compared with the baseline of 56.3. This suggests that attribute information exists in the representation but is not explicitly organized for direct attribute prediction.

The limitation is particularly important for long-tail items, which receive substantially less interaction supervision. In our catalog, 15.1% of items occur at most five times in the training data, while 23.8% never appear in a color-specific conversation, leaving them with little or no direct color supervision.

As a result, a model may learn to generate plausible semantic IDs while failing to preserve fine-grained attributes needed to distinguish similar items.

These observations motivate an additional source of **attribute-specific supervision**. Rather than relying solely on semantic-ID prediction to implicitly encode every item property, we explicitly encourage the model’s hidden representations to preserve fine-grained attribute information. Using color as a concrete instantiation, we apply auxiliary supervision to the item’s catalog-derived CIE-LAB representation. As shown in Figure 3, this supervision primarily benefits data-sparse items: the mean top-1 color error decreases from 54.5 to 50.2 for tail colors and from 33.5 to 29.0 for torso colors, while head colors show essentially no improvement (12.5 to 12.6). This suggests that attribute supervision provides complementary information precisely where interaction-based supervision is weakest. Importantly, the formulation is not specific to color and can be instantiated with other item attributes, such as material, fit, or pattern.

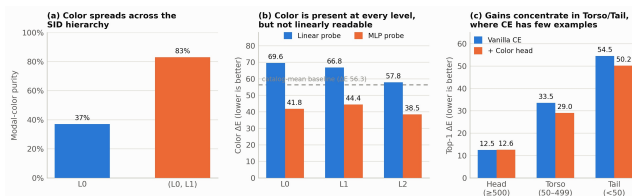


Figure 3: Motivation for attribute-specific supervision. (a) Color purity increases across SID levels, revealing fine-grained attribute structure. (b) Nonlinear probes decode color better than linear probes, indicating latent but unorganized attribute information. (c) Attribute supervision mainly reduces color prediction error for torso and tail items with sparse interaction data.

Color provides a concrete example of this limitation. In our catalog, each item is associated with a CIE-LAB color representation, allowing us to quantify perceptual similarity between items. As shown in Figure 3, color information is distributed across the SID

5 Methodology

We propose **HASTE**, a unified fine-tuning framework that improves semantic-ID-based generative recommendation through two complementary forms of supervision. Given a user interaction context, the model autoregressively generates an item’s semantic ID as a sequence of discrete codes obtained through residual quantization. While standard causal cross-entropy provides supervision for predicting these codes, it treats the semantic-ID sequence uniformly and does not explicitly account for its hierarchical structure. Moreover, item-level attributes provide complementary semantic information that can be particularly valuable for learning representations of data-sparse tail items. Figure 1 provides an overview of the proposed training framework.

HASTE introduces two auxiliary objectives to address these limitations. The first is a **Hierarchical Loss**, which explicitly exploits the multi-level structure of semantic IDs and provides level-aware supervision during semantic-ID generation. By aligning the training objective with the hierarchical organization of the semantic-ID space, the model can learn coarse-to-fine item semantics more

effectively. The second is an **Attribute-Specific Loss**, which introduces auxiliary supervision from fine-grained item attributes. The loss encourages intermediate model representations to retain attribute-level information that may not be sufficiently captured by the semantic-ID prediction objective alone. This additional supervision is particularly beneficial for tail items, where limited interaction data makes it difficult to learn informative representations solely from the primary recommendation objective.

The overall HASTE objective combines the standard causal language-model loss with these two complementary sources of supervision:

$$\mathcal{L}_{\text{HASTE}} = \mathcal{L}_{\text{hier}} + \lambda_{\text{attr}} \mathcal{L}_{\text{attr}}, \quad (5)$$

where λ_{attr} controls the contribution of the attribute-specific objective. The hierarchical loss serves as the primary semantic-ID prediction objective, replacing the standard causal cross-entropy loss, while the attribute loss provides complementary supervision for learning fine-grained item representations. Both the backbone and the auxiliary attribute prediction head are jointly optimized.

5.1 Hierarchical SID Loss

Let an item be represented by a four-level Semantic ID, $\mathbf{y} = (y_1, y_2, y_3, y_4)$, where $y_\ell \in V_\ell$ denotes the code at hierarchy level ℓ , ordered from coarse to fine. The model autoregressively predicts each SID token, with y_2 conditioned on y_1 , with y_3 conditioned on (y_1, y_2) and with y_4 conditioned on (y_1, y_2, y_3) . Unlike standard token-level cross-entropy, which treats all SID positions equally, we introduce a hierarchy-aware loss that accounts for both the structural importance of each level and the model's confidence in its upstream predictions.

Hierarchical weighting. We assign decreasing base weights to successive SID levels using a simple linear schedule,

$$(w_1, w_2, w_3, w_4) = (4, 3, 2, 1), \quad (6)$$

which provides a moderate bias toward coarser levels while retaining supervision at finer levels. Errors at coarser levels can redirect generation to an incorrect subtree and therefore affect downstream predictions; accordingly, they receive larger weights. The root level L_1 is not gated since it has no upstream SID token.

Confidence-aware conditional weighting. For each SID position t , we compute the model's confidence in the ground-truth token as

$$p_t = \left[\max(\text{softmax}(\mathbf{z}_t)_{y_t}, \epsilon) \right]^\gamma, \quad (7)$$

where $\mathbf{z}_t$ denotes the model logits, ϵ is a minimum confidence floor, and γ controls the sharpness of the confidence signal. The confidence is detached from the computation graph to prevent the model from manipulating its loss weight.

The confidence gates for the four hierarchy levels are defined as

$$g_1 = 1, \quad g_2 = p_1, \quad g_3 = p_1 p_2, \quad g_4 = p_1 p_2 p_3. \quad (8)$$

Thus, supervision at a downstream level is strengthened when the model is confident in the preceding decisions, while the confidence floor ensures that a minimum learning signal is retained when upstream predictions are uncertain.

Annealed hierarchical-to-conditional weighting. To avoid relying on unreliable confidence estimates at the beginning of training, we gradually transition from static hierarchical weighting to confidence-aware weighting. Let α be linearly annealed from 1 to 0. The effective weight for level ℓ is

$$\tilde{w}_\ell = w_\ell [\alpha + (1 - \alpha)g_\ell]. \quad (9)$$

At the beginning of training ($\alpha = 1$), the loss uses only the stable hierarchical weights. As training progresses, the contribution of the confidence gate increases, reaching fully conditional weighting when $\alpha = 0$.

Scheduled sampling. Autoregressive SID generation introduces a train–inference mismatch: during teacher forcing, downstream levels are conditioned on ground-truth upstream tokens, whereas inference conditions on the model's own predictions. We reduce this mismatch through scheduled sampling. With probability p_{ss} , the ground-truth L_1 token is replaced by a token sampled from the model's predicted distribution. The sampling probability is linearly increased from zero to a maximum value over the scheduled-sampling annealing period.

The sampled prefix is then used by the subsequent loss-computing forward pass. Consequently, the model is exposed to the types of upstream prediction errors encountered during autoregressive generation, rather than being trained exclusively on clean, teacher-forced trajectories.

Overall objective. The resulting hierarchical SID loss is

$$\mathcal{L}_H = \frac{\sum_t \tilde{w}_t \mathcal{L}_{\text{CE}}(t) \mathbf{1}[\text{valid}(t)]}{\sum_t \mathbf{1}[\text{valid}(t)]}, \quad (10)$$

where $\tilde{w}_t$ is the effective weight associated with the SID level of position t , and $\mathbf{1}[\text{valid}(t)]$ masks non-label positions.

Together, these components provide structure- and confidence-aware supervision: coarse SID decisions receive greater emphasis, while fine-grained supervision is progressively conditioned on the model's ability to resolve upstream semantic decisions. Since both confidence estimation and the final loss are computed on the same (possibly perturbed) input trajectory, the resulting weighting remains consistent with the conditioning context seen by the model.

5.2 Attribute-Conditioned Auxiliary Losses

Hierarchical supervision encourages the model to preserve the coarse-to-fine structure of Semantic IDs (SIDs), but it does not explicitly constrain the learned representations to retain fine-grained item attributes. This is particularly important for attributes such as color and material composition, which may be weakly supervised by conversational interactions, especially for infrequently occurring items. To address this limitation, we introduce a generic attribute-aware auxiliary objective that encourages the hidden representation used for fine-grained SID prediction to preserve attribute information. The proposed objective is not specific to any particular attribute and can be instantiated for any attribute that can be represented by a suitable prediction target. We demonstrate its effectiveness using two complementary item attributes, color and material composition.

Table 1: Performance comparison across different recommendation methods. The best result is shown in bold and the second-best result is underlined.

Dataset	Metric	Sequential		Generative		LLM-Based		Ours (CE loss)	Ours (Hier loss)
		SASRec [15]	BERT4Rec [29]	TIGER [26]	HSTU [40]	IDGenRec [30]	OneRec-Think [20]		
Beauty	Recall@5	0.0402	0.0232	0.0405	0.0424	0.0484	<u>0.0563</u>	0.0325	0.0588 (+4.4%)
	Recall@10	0.0607	0.0396	0.0623	0.0652	0.0693	<u>0.0791</u>	0.0672	0.0823 (+4.0%)
	NDCG@5	0.0254	0.0146	0.0267	0.0280	0.0337	0.0398	0.0326	<u>0.0387</u> (-2.8%)
	NDCG@10	0.0320	0.0199	0.0337	0.0353	0.0404	<u>0.0471</u>	0.0358	0.0503 (+6.8%)
Sports	Recall@5	0.0199	0.0102	0.0215	0.0268	0.0270	<u>0.0288</u>	0.0249	0.0322 (+11.8%)
	Recall@10	0.0301	0.0175	0.0347	0.0343	0.0388	<u>0.0412</u>	0.0327	0.0463 (+12.4%)
	NDCG@5	0.0106	0.0065	0.0137	0.0173	0.0185	0.0199	0.0131	<u>0.0188</u> (-5.5%)
	NDCG@10	0.0141	0.0088	0.0179	0.0226	0.0223	<u>0.0239</u>	0.0212	0.0279 (+16.7%)
Toys	Recall@5	0.0448	0.0215	0.0337	0.0366	<u>0.0595</u>	0.0579	0.0396	0.0655 (+10.1%)
	Recall@10	0.0626	0.0332	0.0547	0.0566	<u>0.0800</u>	0.0797	0.0571	0.0924 (+15.5%)
	NDCG@5	0.0300	0.0131	0.0209	0.0245	<u>0.0432</u>	0.0412	0.0272	0.0479 (+10.9%)
	NDCG@10	0.0358	0.0168	0.0276	0.0309	0.0498	0.0482	0.0328	<u>0.0492</u> (-1.2%)

Auxiliary prediction heads. For each attribute a , we attach a light-weight MLP regression head $[f_a : \mathbb{R}^d \rightarrow \mathbb{R}^{d_a}]$, with architecture $\text{Linear}(d, 64) - \text{GELU} - \text{Linear}(64, d_a)$, where d is the backbone hidden dimension and d_a is the dimensionality of the attribute target. The head operates on the final-layer hidden state at the position used to predict the third-level SID token L_2 . At this position, the model has already conditioned on the preceding coarse-grained codes L_0 and L_1 , making it a natural location for capturing fine-grained perceptual and compositional properties of the item.

SID-conditioned attribute targets. Attribute targets are constructed independently of the SID token vocabulary. We first join the production catalog, which associates each item with its complete SID tuple (l_0, l_1, l_2, l_3) , with the corresponding raw attribute annotations. Since multiple items may share the same SID tuple, we aggregate the attribute annotations of all items assigned to a tuple to obtain a lookup function $[T_a(l_0, l_1, l_2, l_3) \rightarrow y_a]$. During training, the attribute target is therefore determined directly from the item’s complete SID, rather than from the conversation or the surface form of the user query. This allows the auxiliary objective to be applied to every valid SID-generation example, including examples whose conversational context does not explicitly mention the attribute.

Position-aligned supervision. The attribute head operates on the final-layer hidden state at the L_2 SID prediction position, which is the same representation used by the backbone to predict the fine-grained SID token after conditioning on the preceding L_0 and L_1 codes. We choose this position because the finer-grained information captured at the L_2 level is expected to be more closely related to perceptual and compositional item attributes. As discussed in Section 4, our preliminary probing experiments provide empirical evidence that the L_2 representation contains substantially more attribute information than earlier SID positions. Thus, aligning the auxiliary supervision with the L_2 prediction position directly encourages the representation used for fine-grained item prediction

to preserve attribute-specific information.

$$\mathcal{L}_{\text{HASTE}} = \mathcal{L}_{\text{gen}} + \lambda_a \mathcal{L}_a,$$

where $\mathcal{L}_{\text{gen}}$ denotes the generative SID objective and λ_a controls the strength of attribute supervision.

The auxiliary heads are applied to all eligible SID-generation examples rather than only attribute-specific conversations. Because the target is obtained through the SID-to-attribute lookup, the supervision is independent of task type and conversational wording. This substantially increases the density of attribute supervision and is particularly beneficial for items that occur infrequently in attribute-specific interactions.

Attribute-specific instantiations. We instantiate the framework for two complementary item attributes: color and material composition.

Color. For color, the auxiliary head predicts the item’s CIELAB [4] color representation, normalized as

$$\left[\frac{L^*}{50}, \frac{a^*}{80}, \frac{b^*}{80} \right].$$

and use a Huber loss with $\delta = 0.1$. A preliminary frozen-backbone probe study identified the L_2 prediction position as the most informative attachment point, substantially reducing color error compared with earlier SID positions. The same study showed that a nonlinear MLP probe outperforms a linear probe, suggesting that color information is encoded in the representation but is not linearly organized.

To further address the limited supervision available for infrequent items, we apply frequency-aware weighting to the color loss. For an item with training frequency f_i , its auxiliary loss is scaled by

$$r_i = \text{clamp} \left(\left(\frac{f_{\text{med}}}{f_i} \right)^\gamma, 1, 5 \right), \quad \gamma = 0.5$$

where f_{med} is the median item frequency. Thus, low-frequency items receive stronger attribute supervision while preventing excessively large per-example weights.

Material. For material, the target is a 15-dimensional composition vector consisting of 14 named fiber categories and an “other” category. We parse free-text material annotations using a rule-based normalization procedure that incorporates explicit percentages and distributes remaining mass across unweighted material mentions, followed by ℓ_1 normalization: $[\sum_{k=1}^{15} y_{\text{mat},k} = 1.]$ The resulting targets lie in $[0, 1]$, so no additional normalization is required. We use the same Huber objective with $\delta = 0.1$.

Decoupling from generation. The auxiliary objectives do not modify the SID vocabulary, logits, or token-prediction distribution. Instead, they impose additional supervision on the shared hidden representation through lightweight task-specific heads. Consequently, the generative objective remains unchanged while the backbone is encouraged to encode attribute structure that may otherwise be weakly supervised by interaction data. The auxiliary heads are maintained separately from the backbone checkpoint and saved with their architecture metadata and parameters.

6 Experiments

Implementation Details. Each semantic ID consists of four tokens (L_0, L_1, L_2, L_3), with a 256-entry RQ-VAE codebook at each level. Our hierarchical loss uses level-specific base weights (4.0, 3.0, 2.0, 1.0), confidence-based gating, an annealed transition from static weighting to confidence gating, and scheduled sampling to reduce the train–inference exposure gap. For attribute supervision, we use a CIE-LAB distance-based soft-label objective at L_2 , restricted to candidates sharing the ground-truth (L_0, L_1) prefix, and apply it to color-conditioned conversations with weight 0.1. We further employ a hidden-state regression head to provide complementary attribute supervision. Training is performed in two stages, with embedding warm-up followed by full fine-tuning using learning rate 2×10^{-5} , effective batch size 32, and a cosine learning-rate schedule. We train the Qwen3-8B model in two stages. Stage 1 extends the model vocabulary with semantic-ID (SID) tokens and initializes their embeddings. Stage 2 performs full-parameter supervised fine-tuning (SFT), initialized from the Stage 1 checkpoint. All experiments were conducted on NVIDIA RTX6000 single GPU instance with 96 GB memory. Additional implementation details and hyperparameters are provided in the Supplementary Material.

Datasets. We evaluate HASTE across diverse recommendation domains and data regimes. We use four public domains from the Amazon Reviews benchmark [13] – Sports & Outdoors (Sports), Beauty, Toys & Games (Toys), and Fashion-140k, a 140K-item subset of the larger Amazon Fashion dataset, and an internal production-scale proprietary fashion dataset (internal-fashion). Sports, Beauty, and Toys follow the standard P5 5-core protocol [10], where each user and item has at least five interactions, and are used to evaluate the effectiveness of hierarchical supervision for sequential generative recommendation. Fashion-140k and internal-fashion dataset provide richer item-level attribute annotations, including color and material, enabling evaluation of the proposed attribute-specific supervision, particularly in the long-tail setting. Across datasets, item representations are encoded as hierarchical semantic IDs using a

3-level RQ-VAE with 256 codes per level, with a deduplication token appended where applicable. Further details on dataset statistics, preprocessing, semantic-ID construction, and attribute availability are provided in the supplementary material.

Baselines. We compare HASTE against representative baselines spanning three categories: traditional sequential recommendation methods (BERT4Rec [29], SASRec [15]), generative recommendation methods (TIGER [26], HSTU [40]), and LLM-based methods (ID-GenRec [30], OneRec-Think [20]). We additionally compare against the standard cross-entropy (CE) objective to isolate the impact of our proposed supervision.

Evaluation Protocol. We use Recall@K and NDCG@K with $K \in \{5, 10\}$ to evaluate recommendation performance. We evaluate the hierarchical loss on the sequential recommendation task, while attribute-specific losses are evaluated on attribute-conditioned product recommendation, both overall and on torso and tail items with fewer than 500 instances.

6.1 Results

We compare HASTE against representative sequential, generative, and LLM-based recommendation baselines, and evaluate the contribution of its hierarchical and attribute-specific supervision through complimentary experiments.

Overall recommendation performance across Beauty, Sports, and Toys. Table 1 compares HASTE with baseline methods across Beauty, Sports, and Toys datasets. HASTE with hierarchical supervision consistently achieves strong performance, particularly on Recall@10, improving over the strongest baseline by 4.0%, 12.4%, and 15.5% on Beauty, Sports, and Toys, respectively. Similar gains are observed for NDCG@10, with improvements of 6.8%, 16.7%, and 1.2%. The stronger gains on Sports and Toys suggest that hierarchical supervision effectively captures the structured relationships encoded by semantic IDs. Compared with the CE-based variant, hierarchical supervision generally provides further improvements, demonstrating the benefit of explicitly supervising intermediate semantic-ID levels.

Tail-item and attribute-aware performance. Table 2 evaluates hierarchical and attribute-specific supervision on Amazon Fashion-140k and the production-scale proprietary internal-fashion dataset. We use these datasets for attribute losses because Toys, Sports, and Beauty lack the required item-level attribute annotations. Hierarchical supervision consistently improves over CE, with larger gains on tail items, while color and material supervision provide further improvements, particularly for tail items. Under HL, color supervision gives the strongest gains, reaching 24.2% in Recall@10 on Fashion-140k and 17.8% on internal-fashion, while achieving the best overall and tail performance on internal-fashion. These results show that hierarchical and attribute-specific supervision are complementary: the former improves semantic retrieval, while the latter preserves fine-grained attribute information valuable for tail items.

Training efficiency. Table 3 compares the optimization speed of standard causal CE and HL across three Amazon datasets. We define convergence as reaching the same validation-loss level achieved by the corresponding CE model, rather than directly comparing the raw CE and HL loss values. HL consistently reaches this target in

Table 2: Overall and tail-item recommendation performance under hierarchical and attribute-specific supervision. Values in blue denote relative improvement over the corresponding baseline: CE for HL and CE-based attribute variants, and HL for HL-based attribute variants. Bold values indicate the best performance, while underlined values indicate the second-best performance.

Method	Split	Amazon Fashion-140k				Internal-fashion			
		Recall		NDCG		Recall		NDCG	
		@5	@10	@5	@10	@5	@10	@5	@10
CE	Overall	0.0243	0.0324	0.0236	0.0452	0.0255	0.0319	0.0376	0.0425
	Tail	0.0134	0.0169	0.0245	0.0298	0.0267	0.0342	0.0387	0.0393
HL	Overall	0.0297 (+22.2%)	0.0345 (+6.5%)	0.0316 (+33.9%)	0.0486 (+7.5%)	0.0284 (+11.4%)	0.0351 (+10.0%)	0.0412 (+9.6%)	0.0468 (+10.1%)
	Tail	0.0168 (+25.4%)	0.0215 (+27.2%)	0.0297 (+21.2%)	0.0354 (+18.8%)	0.0301 (+12.7%)	0.0387 (+13.2%)	0.0435 (+12.4%)	0.0459 (+16.8%)
CE + Color Loss	Overall	0.0256 (+5.3%)	0.0335 (+3.4%)	0.0251 (+6.4%)	0.0464 (+2.7%)	0.0270 (+5.9%)	0.0331 (+3.8%)	0.0391 (+4.0%)	0.0438 (+3.1%)
	Tail	0.0167 (+24.6%)	0.0218 (+29.0%)	0.0298 (+21.6%)	0.0361 (+21.1%)	0.0319 (+19.5%)	0.0412 (+20.5%)	0.0461 (+19.1%)	0.0487 (+23.9%)
HL + Color Loss	Overall	0.0305 (+2.7%)	0.0353 (+2.3%)	0.0325 (+2.8%)	0.0495 (+1.9%)	0.0291 (+2.5%)	0.0359 (+2.3%)	0.0423 (+2.7%)	0.0476 (+1.7%)
	Tail	0.0202 (+20.2%)	0.0267 (+24.2%)	0.0358 (+20.5%)	0.0431 (+21.8%)	0.0348 (+15.6%)	0.0456 (+17.8%)	0.0517 (+18.9%)	0.0549 (+19.6%)
CE + Material Loss	Overall	0.0252 (+3.7%)	0.0332 (+2.5%)	0.0248 (+5.1%)	0.0461 (+2.0%)	0.0268 (+5.1%)	0.0328 (+2.8%)	0.0389 (+3.5%)	0.0435 (+2.4%)
	Tail	0.0161 (+20.1%)	0.0209 (+23.7%)	0.0287 (+17.1%)	0.0348 (+16.8%)	0.0310 (+16.1%)	0.0401 (+17.3%)	0.0450 (+16.3%)	0.0478 (+21.6%)
HL + Material Loss	Overall	<u>0.0303</u> (+2.0%)	<u>0.0351</u> (+1.7%)	<u>0.0322</u> (+1.9%)	<u>0.0492</u> (+1.2%)	<u>0.0289</u> (+1.8%)	<u>0.0357</u> (+1.7%)	<u>0.0420</u> (+1.9%)	<u>0.0473</u> (+1.1%)
	Tail	<u>0.0195</u> (+16.1%)	<u>0.0258</u> (+20.0%)	<u>0.0349</u> (+17.5%)	<u>0.0419</u> (+18.4%)	<u>0.0341</u> (+13.3%)	<u>0.0447</u> (+15.5%)	<u>0.0508</u> (+16.8%)	<u>0.0538</u> (+17.2%)

Table 3: Training efficiency of standard causal cross-entropy (CE) and Hierarchical Loss (HL). HL substantially reduces the number of training steps required to reach convergence.

Dataset	CE Steps	HL Steps	Speedup
Toys	92000	42000	2.19 ×
Sports	100000	53000	1.89 ×
Beauty	85000	44000	1.93 ×
Average	–	–	1.99 ×

substantially fewer training steps, reducing the required steps by more than half on Toys and by roughly half on Sports and Beauty. On average, HL achieves a 1.99× speedup over CE, demonstrating that hierarchical supervision not only improves recommendation quality but also enables more efficient optimization.

Ablation Studies. We conduct ablation on the Toys dataset to examine the contribution of hierarchical supervision and its individual components. Table 4 shows that progressively supervising additional semantic-ID levels consistently improves NDCG@10. Performance increases from 0.0355 with supervision at the first level to 0.0492 when all four levels are supervised, demonstrating that finer-grained semantic codes provide complementary supervision.

Table 5 further isolates the contributions of the hierarchical loss components. Static weighting (SW) assigns fixed weights of 4:3:2:1 to the four SID levels, emphasizing coarser semantic decisions. Conditional gating (CG) uses confidence-aware weights, strengthening downstream supervision when preceding predictions are reliable. Scheduled sampling (SS) gradually replaces the ground-truth L_1 token with model predictions to reduce the teacher-forcing mismatch. Each component improves over causal CE, with CG providing the largest individual gain. The full configuration combines SW, CG,

and SS, yielding the best performance and improving Recall@10 and NDCG@10 by 44.3% and 50.0%, respectively, over causal CE. The substantial improvement from the combined setup suggests that the three mechanisms provide complementary benefits: level-aware weighting, confidence-adaptive supervision, and robustness to autoregressive errors. Further details on the hyperparameter settings used in this ablation are provided in the supplementary material.

Table 4: Ablation of hierarchical supervision across semantic-ID levels on Toys dataset. Using conditional gating, we progressively incorporate higher-level semantic codes into the Hierarchical Loss.

SID Supervision	NDCG@10	Δ vs. CE
Causal CE	0.0328	–
(s_a)	0.0355	+0.0027
(s_a, s_b)	0.0410	+0.0082
(s_a, s_b, s_c)	0.0435	+0.0107
(s_a, s_b, s_c, s_d)	0.0492	+0.0164

7 Conclusion

We presented HASTE, a unified framework for generative recommendation with hierarchical and attribute-aware supervision. By exploiting the multi-level structure of semantic IDs and incorporating confidence-aware gating, level weighting, and scheduled sampling, HASTE improves hierarchical prediction. Attribute-specific auxiliary objectives further preserve fine-grained item information, particularly for tail items. Experiments across multiple Amazon domains and a large-scale fashion dataset show consistent improvements over strong baselines, with ablations confirming the complementary benefits of the proposed components.

Table 5: Ablation of individual components of the Hierarchical Loss on the Toys dataset. Here, *HL* denotes Hierarchical Loss, *SW* static level weighting, *CG* conditional gating, and *SS* scheduled sampling.

Supervision	Recall@10	NDCG@10	Δ vs. CE (%)
CE	0.0571	0.0328	–
HL w/ SW	0.0624	0.0386	+9.3 / +17.7
HL w/ CG	0.0678	0.0402	+18.7 / +22.6
HL w/ SS	0.0598	0.0357	+4.7 / +8.8
HL w/ SW, CG, SS	0.0824	0.0492	+44.3 / +50.0

Some limitations of our work include its dependence on the quality of the underlying semantic-ID hierarchy and the availability of reliable attribute information for attribute-specific supervision. Moreover, our evaluation primarily relies on offline metrics, which may not fully capture real-world user satisfaction and long-term engagement. Future work will explore adaptive hierarchy construction, richer multimodal and behavioral attribute supervision, and online evaluation incorporating user feedback and long-term engagement.

8 Ethical Considerations

HASTE is designed to improve generative recommendation and does not directly use sensitive user attributes. However, recommendation systems can amplify existing popularity biases and may disproportionately favor frequently interacted-with items, potentially reducing exposure to less popular or tail items. HASTE explicitly incorporates attribute-aware supervision to improve tail-item representation, but this does not guarantee fairness across all products or user groups. In addition, the use of catalog and behavioral data may introduce privacy concerns depending on the data collection and deployment setting. In practice, appropriate data anonymization, access controls, and privacy-preserving data practices should be applied. Finally, the framework should be evaluated for unintended recommendation biases before deployment, particularly in settings where recommendations can influence user choices or commercial outcomes.

9 Appendix

9.1 Dataset Details

We evaluate HASTE on five datasets spanning public Amazon recommendation [13] and a production-scale internal fashion dataset: Sports & Outdoors (Sports), Beauty, Toys & Games (Toys), Fashion-140k, and Internal-fashion. Table 6 summarizes their catalog sizes, user sequences, interaction volumes, conversational training data, and available metadata. Detailed preprocessing, semantic-ID construction, and attribute availability are described below.

9.1.1 Sports, Beauty, and Toys.

Source and preprocessing. Sports, Beauty, and Toys use the Amazon-2014 5-core P5 release [10]. Each category contains one row per

ASIN, while user interaction sequences are the chronological P5 5-core sequences, retaining users with at least five reviews in the corresponding category. The resulting catalogs contain 18,357, 12,101, and 11,924 items for Sports, Beauty, and Toys, respectively.

Table 7 provides metadata coverage. Approximately 0.5% of Sports and Toys items have no title in the upstream metadata (Product: N/A); these items produce near-degenerate embeddings and account for a disproportionate share of semantic-ID collisions. Toys has depth-3 taxonomy labels for only 28.6% of items; therefore, statistics conditioned on this taxonomy level apply only to that subset.

Evaluation split. The datasets are prepared using a leave-one-out (LOO) strategy for sequential evaluation. For each user sequence, the last item is held out for testing, the item immediately preceding it is used for validation, and all earlier items are used for training. Thus, each sequence contributes a training history, a validation target, and a test target while preserving chronological order. For the conversational SFT corpus, examples are independently shuffled and split 92%/5%/3% into training, validation and test set. Sequences use a leave-two-out protocol. For each user, the training example uses $\text{seq}[: -2]$, validation predicts $\text{seq}[-2]$ from $\text{seq}[: -2]$, and test predicts $\text{seq}[-1]$ from $\text{seq}[: -1]$.

Metadata and attributes. In addition to title, description, category, and brand, the Amazon categories provide price, sales rank, and related-item information. They do not contain the dense structured attributes required by our attribute-specific objectives, such as fine-grained color, material, style, and occasion. Consequently, these datasets are used primarily to evaluate hierarchical supervision.

9.1.2 Fashion-140k.

Source and preprocessing. Fashion-140k contains 141,300 items selected from an internally processed 799,953-item Amazon Fashion pool. The subset is enriched with Gemini-extracted structured attributes that are not available in the raw Amazon metadata. User sequences are obtained using 5-core filtering on Amazon Fashion reviews, restricted to ASINs in the item pool. Sequences shorter than three items are removed and sequences are capped at 100 items, resulting in 21,326 sequences with mean length 3.33, median length 3, and maximum length 15.

Evaluation split. Sequences use a leave-two-out protocol. For each user, the training example uses $\text{seq}[: -2]$, validation predicts $\text{seq}[-2]$ from $\text{seq}[: -2]$, and test predicts $\text{seq}[-1]$ from $\text{seq}[: -1]$. All 21,326 users are retained across the splits.

Structured attributes. Fashion-140k provides dense structured attributes, including product type, gender, brand, occasion, primary and secondary color, style, material, and fit. Their coverage is summarized in Table 8.

9.1.3 Internal-fashion.

Source and preprocessing. Internal-fashion is a production-scale proprietary fashion engagement dataset containing 163,724 products, 103,047 qualifying user sequences, and 1,640,953 interaction events. The dataset is constructed from raw interaction logs joined with persona/location metadata, user-profile information, and the

Table 6: Dataset statistics. We report catalog size, user sequences, interaction events, conversation statistics, and metadata attributes. “–” indicates that the corresponding statistic is not applicable.

Statistic	Toys	Beauty	Sports	Fashion-140k	Internal-fashion
Items (catalog)	11,924	12,101	18,357	141,300	163,724
User sequences	19,412	22,363	35,598	21,326	103,047
Total events	167,597	198,502	296,337	71,059	1,640,953
Seq. length (min/med/mean/max)	5/6/8.63/550	5/6/8.88/204	5/6/8.32/296	3/3/3.33/15	3/6/15.92/100
Total conversations	1,390,005	1,720,660	2,508,966	1,407,009	103,047
Train / Validation / Test	1,278,805/69,500/19,412	1,583,007/86,033/22,363	2,308,249/125,448/35,598	1,294,448/70,350/21,326	6,255,087/339,950/203,971
Conversation types	22	26	26	40	40
Raw metadata attributes	9	9	9	21	20

Table 7: Item and metadata coverage for Sports, Beauty, and Toys.

	Sports	Beauty	Toys
Items	18,357	12,101	11,924
Users (sequences)	35,598	22,363	19,412
Title coverage	99.5%	99.9%	99.5%
Description coverage	90.5%	98.3%	98.5%
Brand coverage	67.2%	82.8%	85.6%
Price coverage	80.4%	95.2%	92.9%
Taxonomy depth-3 coverage	—	—	28.6%

Table 8: Structured-attribute coverage in Fashion-140k.

Attribute	Count	Coverage
Items	141,300	—
Sequences	21,326	—
product_type	141,220	99.9%
gender	138,332	97.9%
brand	130,177	92.1%
occasion	130,066	92.0%
primary_color	129,772	91.8%
style	125,727	89.0%
material	90,224	63.9%
secondary_color	64,972	46.0%
fit	36,212	25.6%

product catalog. Six deliberate-intent engagement types are retained: BUY, LIKE, SHARE, DOWNLOAD, TRY_ON, and SAVE. TAP events are excluded because they represent a UI prerequisite rather than explicit engagement; including them would add approximately 16.5M events and cause 17% of users to exceed the 100-item sequence cap.

Users must have a resolved city or country in the persona metadata. User identity is represented by the pair (user_id, profile_id), and users mapping to multiple profiles are removed. Both fully specified and partially specified persona contexts are retained during training.

Deduplication and sequence construction. Three deduplication passes remove system-generated duplicate events while preserving genuine repeated actions: ambiguous sessions containing multiple user-profile combinations are removed, exact row duplicates

are discarded, and interaction-level duplicates are removed only when they match simultaneously under both session/position and timestamp-based windowing criteria. Interactions are then grouped by user-profile pair and ordered chronologically. Sequences have a minimum length of three and are capped at 100 items. Consecutive events separated by 30 days or more are discarded.

Training-pair generation. Six interaction-pattern families generate conversational training examples: cross-product transitions, same-product persistence, forward engagement, persona-conditioned prediction, backward engagement, and cross-signal escalation. Temporal gaps are bucketed into same-session (<30 min), within-day (<24h), within-week (<7d), and within-month (7–30d) intervals. This process produces 6,799,008 conversations, split into 6,255,087 training, 339,950 validation, and 203,971 test examples.

Attributes. The Internal-fashion catalog provides brand, price, currency, gender, color, size, materials, category name, taxonomy, and VLM/attribute-assembled product descriptions. This attribute richness makes Internal-fashion, together with Fashion-140k, the primary setting for evaluating attribute-specific supervision.

9.1.4 Semantic-ID Construction. We construct semantic IDs using a TIGER-style [26] residual vector quantization autoencoder (RQ-VAE). For Sports, Beauty, and Toys, title, description, category path, and brand are concatenated into a single context string and encoded into a 1024-dimensional representation. Fashion-140k uses the RQ-VAE checkpoint trained on the full 799,953-item Amazon Fashion pool; semantic IDs for its 141,300 items are obtained by nearest-codebook assignment in the same embedding space. Internal-fashion additionally incorporates multimodal product representations using 1024-dimensional Qwen3-VL embeddings.

The RQ-VAE encoder follows a $1024 \rightarrow 512 \rightarrow 256 \rightarrow 128 \rightarrow 32$ architecture and uses three residual codebooks with 256 entries each. For Sports, Beauty, and Toys, the RQ-VAE is trained for 20,000 steps per category on a single A100 GPU. A fourth deduplication token is appended to the three quantized codes when collisions occur, producing the final depth-4 semantic ID used for language-model fine-tuning.

For Sports, Beauty, and Toys, the unique three-level SID rates are 94.6%, 96.3%, and 97.4%, respectively, corresponding to collision rates of 5.4%, 3.7%, and 2.6%. The maximum deduplication indices observed are 13, 35, and 11. All 256 codes are utilized at each quantization level. Fashion-140k has a higher collision rate because its items are quantized using a codebook trained on the

Table 9: Semantic-ID construction and representation statistics across datasets. Each dataset uses three residual quantization levels with 256 codes per level; a deduplication token is appended when required.

Statistic	Toys	Beauty	Sports	Fashion-140k	Internal-fashion
Representation	1024-d text			1024-d text	1024-d multimodal
Unique 3-tuple SIDs	11,615	11,649	17,361	127,844	153,046
Collision rate	2.6%	3.7%	5.4%	9.5%	6.5%

larger 799,953-item pool rather than directly on the subset. The deduplication token therefore provides a deterministic mechanism for distinguishing items that share the same three-level semantic code.

9.1.5 Training-Data Construction.

Hierarchical-loss training. The hierarchical loss is a training objective rather than a separate data-generation pipeline. It is applied to the same semantic-ID conversational training corpus used by the flat causal CE baseline, reweighting the per-level losses without changing the underlying examples.

For Sports, Beauty, and Toys, conversational SFT data are generated from item metadata, semantic IDs, and user sequences using fixed instruction templates. Sports and Beauty use 26 conversation types, while Toys uses 22 because several types requiring lower-coverage fields are omitted. Repeated instructions mapping to the same SID are intentionally retained to reinforce the SID-item mapping and provide surface-form diversity. Examples are shuffled with a fixed seed and split 92/5/3 into training and validation.

The conversation types fall into five groups:

- **SID→text decoding:** recover title, description, context, category, or brand from a semantic ID.
- **Text→SID encoding:** predict a semantic ID from title, description, or brand information.
- **Sequential next-item prediction:** predict the next item from histories of length 2, 3, 5 and all.
- **Co-engagement:** predict co-purchased, frequently bought, or similar products.
- **Category/prefix structure understanding:** reason over shared semantic-ID prefixes and their associated categories or examples.

This produces 2,383,517, 1,634,627, and 1,320,505 training conversations for Sports, Beauty, and Toys, respectively.

Attribute-loss training. The attribute-conditioned loss is evaluated on Fashion-140k and Internal-fashion, where structured attributes are sufficiently dense. The loss is applied as an auxiliary objective over the training corpus rather than being restricted exclusively to attribute-conditioned conversations.

Fashion-140k. The 1,336,658-example training corpus is organized into five families:

- **AF1 – SID↔text/attribute mappings:** approximately 700K examples covering SID-to-text, text-to-SID, and attribute-to-SID mappings.
- **AF2 – attribute-based retrieval:** approximately 27K examples using gender, product type, color, style, occasion, brand, and material, including single-, two-, and three-attribute queries.

- **AF3 – sequential prediction:** approximately 200K next-item and co-purchase examples.
- **AF4 – outfit color coordination:** approximately 1K examples derived from Fashionpedia and Polyvore color-compatibility rules.
- **AF5 – semantic hierarchy understanding:** approximately 117K examples based on SID-prefix clusters and similar-item relationships.

AF2 and AF4 provide the most direct attribute supervision because their targets are explicitly organized by shared attributes such as color, material, gender, and style.

Internal-fashion. The Internal-fashion SFT corpus contains 368,482 training and 19,482 validation conversations using the same 57-type schema as the underlying pipeline. In addition to SID↔text tasks, it contains exhaustive attribute-conjunction retrieval examples over combinations of brand, color, material, gender, and category. Examples include queries for all products matching a particular color-category or brand-color combination.

Rather than restricting the auxiliary objective to these explicitly color-named examples, the color loss is applied as a soft per-example reweighting over the entire corpus. Examples whose conversation type contains “color” receive the full auxiliary weight ($\alpha = 0.1$), while all other examples receive a reduced scale of 0.2α . Thus, even items that do not occur in explicitly color-conditioned conversations receive a weaker color-anchoring gradient.

The color objective uses soft-label cross-entropy over color-similar candidates, with similarity determined by $\exp(-\Delta E/T)$ in CIE-LAB [4] color space and an additional frequency-based weighting term that upweights rare items. This formulation allows attribute supervision to provide stronger fine-grained signals while retaining the broad conversational training distribution.

9.2 Implementation Details

9.2.1 Hierarchical Loss. We provide implementation details for the confidence-gated hierarchical loss used to evaluate Sports, Beauty, and Toys. The loss combines static level weighting, confidence-based gating, annealed blending, and scheduled sampling.

Static level weighting. Each SID level l is assigned a fixed base weight w_l , with larger weights for coarser levels. Non-SID tokens retain a weight of 1.0. This provides hierarchical supervision from the beginning of training, independently of the model’s confidence.

Confidence-based gating. For each SID level $l > 0$, the base weight is modulated by the model’s confidence in the correct codes at all preceding levels. Specifically, let $p_{\text{correct}}(s_{i,k})$ denote the predicted probability of the ground-truth code $s_{i,k}$ at level k . The effective

weight is

$$w_l^{\text{eff}} = w_l \left[\alpha + (1 - \alpha) \prod_{k < l} p_{\text{correct}}(s_{i,k})^\gamma \right], \quad (11)$$

where

$$p_{\text{correct}}(s_{i,k}) \leftarrow \text{clamp}(p_{\text{correct}}(s_{i,k}), \min = \text{floor}). \quad (12)$$

The confidence values are detached from the computation graph, preventing the model from increasing the gate through the gating pathway itself. We use a gating floor of 0.2 and $\gamma = 1.0$.

Table 10: Hyperparameter settings for the hierarchical loss. Base weights are ordered from the coarsest to the finest SID level. The same configuration is used across Sports, Beauty, and Toys.

Hyperparameter	Setting
Base weights (w_0, w_1, w_2, w_3)	(4.0, 3.0, 2.0, 1.0)
Gating floor	0.2
Gating sharpness γ	1.0
Max scheduled-sampling rate	0.3
Scheduled-sampling anneal steps	200,000
Blend anneal steps	300,000
Learning rate	2×10^{-5}
Per-device batch size	4
Gradient accumulation	8 (effective batch size 32)
Epochs	20
LR schedule	Cosine, 3% linear warmup
Weight decay	0.01
Gradient clip norm	1.0
Optimizer	adamw_8bit
Maximum sequence length	2048 tokens

Annealed confidence blending. The coefficient α controls the transition from static weighting to confidence-based weighting. It is initialized to 1.0, corresponding to purely static weighting, and linearly decreases to 0.0 over 300,000 optimizer steps. Thus, training initially emphasizes a stable hierarchical curriculum and gradually transitions to adaptive supervision based on the model’s confidence in preceding levels.

Scheduled sampling. We additionally use scheduled sampling to reduce the exposure gap between teacher-forced training and autoregressive inference. A first forward pass, performed without gradient computation, samples the predicted code at the coarsest SID level from the model’s categorical distribution. With probability ss_prob , this prediction replaces the corresponding ground-truth code in the input sequence. A second forward pass then computes the hierarchical loss on the resulting sequence. The scheduled-sampling probability is linearly increased from 0 to 0.3 over 200,000 optimizer steps. Multinomial sampling is used rather than greedy decoding to expose the model to the stochasticity of autoregressive generation.

Category-specific configurations. The hierarchical-loss mechanism is shared across all three datasets, with category-specific differences in the number of SID levels and their base weights. Sports, Beauty and Toys use four SID codes with weights (4.0, 3.0, 2.0, 1.0). The fourth code corresponds to the deduplication code in addition to the three residual-quantization levels. All other hierarchical-loss hyperparameters are shared across datasets, as summarized in Table 10.

Training and inference. Training is performed in two stages. First, only the newly introduced SID embeddings are optimized for 50,000 steps using flat cross-entropy with a learning rate of 1×10^{-3} . The resulting checkpoint initializes the second stage, in which all model parameters are jointly fine-tuned using the hierarchical loss and the configuration in Table 10.

During evaluation, autoregressive generation is constrained to catalog-valid SID sequences using a prefix-tree logits processor constructed from up to 100K training sequences. If a generated prefix is not present in the tree, decoding falls back to a per-position vocabulary mask over the 256 valid SID codes at the corresponding level.

9.2.2 Attribute-Specific Losses.

Shared architecture and attachment. We use the same auxiliary-head architecture for both color and material composition prediction. The head is attached to the last-layer hidden state at the L2 semantic-ID prediction position, h_{L2} , after the model has consumed the preceding L0 and L1 codes. Each head is a two-layer MLP,

$$\text{MLP}(h) = \text{Linear}(4096, 64) \rightarrow \text{GELU} \rightarrow \text{Linear}(64, d_{\text{out}}),$$

where $d_{\text{out}} = 3$ for color and $d_{\text{out}} = 15$ for material composition. The heads are trained jointly with the full LLM backbone. The overall objective is

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_{\text{attr}} \mathcal{L}_{\text{Huber}},$$

where the attribute loss is applied at every SID position in every conversation, without conditioning on conversation type.

Color supervision. Color targets are derived from catalog-level CIE-LAB annotations. For each SID sequence (L_0, L_1, L_2, L_3), we average the LAB values of all catalog items assigned to that triple. Before computing the loss, L^* is normalized as $(L^* - 50)/50$, while a^* and b^* are normalized as $a^*/80$ and $b^*/80$. We use a mean-reduced Huber loss with $\delta = 0.1$:

$$\mathcal{L}_{\text{color}} = \text{Huber}_{0.1}(\hat{\mathbf{y}}_{\text{color}}, \mathbf{y}_{\text{color}}).$$

To emphasize data-sparse semantic IDs, each example is weighted according to its L2-code frequency:

$$w_i = \text{clip} \left(\left(\frac{f_{\text{med}}}{f_i} \right)^{0.5}, 1, 5 \right).$$

The weighted loss is multiplied by $\lambda_{\text{color}} = 0.1$.

Material-composition supervision. Material annotations are converted into 15-dimensional composition vectors comprising 14 canonical material categories plus an *other* category. Explicit percentages are retained, while unspecified components divide the remaining mass uniformly; the resulting vector is L1-normalized to

Table 11: One example per conversation sub-type in the Toys dataset, grouped into the five families of Section 9.1.5. Semantic IDs are abbreviated as {sid_W,X,Y,Z} in this table only, where W, X, Y, and Z denote the four semantic-ID levels.

Group	Sub-type (<i>n</i>)	User turn	Assistant turn
SID → text (decoding)	sid_to_title (11,865)	Product {sid_58,401,722,768} has title:	Doctor Who – Master Laser Screwdriver
	sid_to_description (10,736)	Item {sid_30,341,570,768} is described as:	Only Playhut structures provide children with hours of interactive adventure. EZ Twist technology allows for instant setup and quick fold-down...
	sid_to_context (11,865)	Tell me about product {sid_107,383,713,768}:	Title: Crayola Tub of Fun; Description: Crayola Tub of Fun; Category: Toys & Games
	sid_to_category (11,924)	Product {sid_122,381,749,768} belongs to category:	Toys & Games > Games > Board Games
	sid_to_brand (10,199)	Product {sid_77,477,593,768} is made by:	Parker Brothers
Text → SID (encoding)	title_to_sid (11,865)	The product “Bandai Tamashii Stage Act.5 for Mechanics Clear” has SemanticID:	{sid_56,289,628,768}
	description_to_sid (7,695)	A product described as “Hit the road in style— it’s the coolest way to travel and soak up some sun. This two-seater convertible has sweet details...” has SemanticID:	{sid_176,504,546,768}
	brand_to_sid (10,199)	A Toys & Games product made by “Learning Curve” has SemanticID:	{sid_199,352,680,768}
Sequential next-item prediction	seq_last_2 (128,773)	User’s last purchases: {sid_94,452,645,768}, {sid_246,397,725,768}. Next:	{sid_49,283,602,768}
	seq_last_3 (109,361)	Based on recent purchases: {sid_53,484,691,768}, {sid_83,423,723,768}, {sid_211,471,616,768}, next item:	{sid_12,442,677,768}
	seq_last_5 (70,537)	Purchase sequence: {sid_129,456,766,768}, {sid_6,387,739,768}, {sid_6,302,559,768}, {sid_6,387,591,768}, {sid_79,487,764,768}. Predict next:	{sid_6,387,554,768}
Co-engagement	copurchase_forward (474,246)	A user who bought {sid_167,411,542,768} (Melissa & Doug Sunny Patch Happy Giddy Chair) might also buy:	{sid_210,329,533,768}
	copurchase_backward (472,066)	Before buying {sid_192,300,739,768} (Playskool Learnimals ABC Adventure Elefun Toy), users often buy:	{sid_68,326,561,768}
	similar_products (4,403)	Products similar to {sid_19,273,603,768} include:	{sid_19,273,661,768}
Category / prefix structure understanding	prefix_category (256)	Products starting with {sid_118} are typically:	Toys & Games products
	prefix_2level_category (442)	Products starting with {sid_140,261} are:	Toys & Games items
	prefix_examples (256)	Examples of products with prefix {sid_20}:	{sid_20,374,673,768}, {sid_20,491,661,768}, {sid_20,354,723,768}
	category_transition (8)	After buying from Health & Personal Care, users often buy from:	Toys & Games

sum to one. SID-level targets are obtained by averaging composition vectors across all items sharing the same SID triple. We apply a mean-reduced Huber loss with $\delta = 0.1$ directly in composition space:

$$\mathcal{L}_{\text{material}} = \text{Huber}_{0.1}(\hat{\mathbf{y}}_{\text{material}} \cdot \mathbf{y}_{\text{material}}).$$

Table 12: Hyperparameter settings for the hierarchical-loss component ablation. Base weights are ordered from the coarsest to the finest SID level. All variants use three SID levels.

Variant	Base weights (w_1, w_2, w_3, w_4)	α schedule	Gate floor / γ	ss_max_prob
CE	(1.0, 1.0, 1.0, 1.0)	$\alpha \equiv 1.0$	- / -	0
HL w/ SW	(4.0, 3.0, 2.0, 1.0)	$\alpha \equiv 1.0$	- / -	0
HL w/ CG	(1.0, 1.0, 1.0, 1.0)	1.0 $\rightarrow$ 0.0 over 300,000 steps	0.2 / 1.0	0
HL w/ SS	(1.0, 1.0, 1.0, 1.0)	$\alpha \equiv 1.0$	- / -	0.3
HL w/ SW, CG, SS (full)	(4.0, 3.0, 2.0, 1.0)	1.0 $\rightarrow$ 0.0 over 300,000 steps	0.2 / 1.0	0.3

Unlike color, material supervision uses no frequency-based tail weighting. We set $\lambda_{\text{material}} = 1.0$ to retain sufficient gradient magnitude for the smaller raw composition loss.

Training details. Both auxiliary heads are dynamically attached to the pretrained model and optimized jointly with the full backbone. The attribute loss is evaluated for every SID-bearing training example rather than only attribute-specific conversations. For these experiments, the CE component is computed over the full input sequence (prompt and completion), corresponding to `completion_only_loss=False`. The auxiliary heads are saved separately from the Hugging Face model checkpoint.

9.2.3 Hyperparameter Settings for the Hierarchical-Loss Component Ablation. Table 12 reports the hyperparameter settings used for the hierarchical-loss component ablation. Each variant is derived from the full configuration by disabling the other mechanisms, allowing the contribution of static weighting (SW), confidence gating (CG), and scheduled sampling (SS) to be isolated. The **CE** baseline disables all three mechanisms and applies uniform weighting to all SID tokens. **HL w/ SW** uses static level weights while disabling confidence gating and scheduled sampling. **HL w/ CG** uses confidence-based gating with uniform base weights, while disabling scheduled sampling. **HL w/ SS** uses scheduled sampling with uniform base weights and no confidence gating. **HL w/ SW, CG, SS** corresponds to the full hierarchical-loss configuration.

All variants use the same optimizer and training configuration: learning rate 2×10^{-5} , effective batch size 16, 20 training epochs, and a cosine learning-rate schedule. Scheduled sampling, when enabled, uses an annealing period of 200,000 steps. The static level weights are (3.0, 2.0, 1.0) from the coarsest to the finest level. Confidence gating uses a floor of 0.2, $\gamma = 1.0$, and an α schedule that linearly decreases from 1.0 to 0.0 over 300,000 steps.

9.2.4 Conversation Type Examples (Toys). Table 11 gives one representative example of every conversation sub-type in the Toys dataset ($n = 1,390,005$ total conversations), grouped into the five families described above. Semantic IDs are shown in the model’s native token format, `<|sid_start|><|sid_W|><|sid_X|><|sid_Y|><|sid_Z|><|sid_end|>`, where $W, X, Y, Z \in [0, 255]$ index the four RQ-VAE codebook levels.

References

- [1] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent Neural networks. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1* (Montreal, Canada) (NIPS’15). MIT Press, Cambridge, MA, USA, 1171–1179.
- [2] Luca Bertinetto, Romain Mueller, Konstantinos Tertikas, Sina Samangooei, and Nicholas A Lord. 2020. Making better mistakes: Leveraging class hierarchies with deep networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 12503–12512.
- [3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.
- [4] CIE. 2007. *Colorimetry – Part 4: CIE 1976 L*a*b* Colour Space*. Number ISO 11664-4:2008(E) / CIE S 014-4/E:2007. Commission Internationale de l’Eclairage, Vienna, Austria.
- [5] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks for YouTube Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems* (Boston, Massachusetts, USA) (RecSys ’16). Association for Computing Machinery, New York, NY, USA, 191–198. doi:10.1145/2959100.2959190
- [6] Nicola De Cao, Gautier Izcard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive Entity Retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3–7, 2021*. OpenReview.net. <https://openreview.net/forum?id=5k8F6U039V>
- [7] 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* (2025).
- [8] Chao Feng, Wuchao Li, Defu Lian, Zheng Liu, and Enhong Chen. 2022. Recommender Forest for Efficient Retrieval. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 38912–38924. doi:10.52202/068431-2820
- [9] Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. 2023. Chat-rec: Towards interactive and explainable llms-augmented recommender system. *arXiv preprint arXiv:2303.14524* (2023).
- [10] Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM conference on recommender systems*. 299–315.
- [11] Ruining He, Lukasz Heldt, Lichan Hong, Raghunandan Keshavan, Shifan Mao, Nikhil Mehta, Zhengyang Su, Alicia Tsai, Yueqi Wang, Shao-Chuan Wang, et al. 2026. Plum: Adapting pre-trained language models for industrial-scale generative recommendations. In *Proceedings of the ACM Web Conference 2026*. 8093–8104.
- [12] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).
- [13] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiushi Chen, and Julian McAuley. 2024. Bridging Language and Items for Retrieval and Recommendation. *arXiv preprint arXiv:2403.03952* (2024).
- [14] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2024. Large language models are zero-shot rankers for recommender systems. In *European conference on information retrieval*. Springer, 364–381.
- [15] Wang-Cheng Kang and Julian McAuley. 2018. Self-Attentive Sequential Recommendation. arXiv:1808.09781 [cs.LG]. <https://arxiv.org/abs/1808.09781>
- [16] Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu. 2015. Deeply-Supervised Nets. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 38)*, Guy Lebanon and S. V. N. Vishwanathan (Eds.). PMLR, San Diego, California, USA, 562–570. <https://proceedings.mlr.press/v38/lee15a.html>
- [17] Doyup Lee, Chihyeon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. 2022. Autoregressive image generation using residual quantization. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*. IEEE, 11513–11522.
- [18] Jinming Li, Wentao Zhang, Tian Wang, Guanglei Xiong, Alan Lu, and Gerard Medioni. 2023. GPT4Rec: A generative framework for personalized recommendation and user interests interpretation. *arXiv preprint arXiv:2304.03879* (2023).
- [19] Han Liu, Yinwei Wei, Xueming Song, Weili Guan, Yuan-Fang Li, and Liqiang Nie. 2024. Mmgrec: Multimodal generative recommendation with transformer model. *ACM Transactions on Multimedia Computing, Communications and Applications*

- (2024).
- [20] Zhanyu Liu, Shiyao Wang, Xingmei Wang, Rongzhou Zhang, Jiaxin Deng, Honghui Bao, Jinghao Zhang, Wuchao Li, Pengfei Zheng, Xiangyu Wu, Yifei Hu, Qigen Hu, Xinchun Luo, Lejian Ren, Zixing Zhang, Qianqian Wang, Kuo Cai, Yunfan Wu, Hongtao Cheng, Zexuan Cheng, Lu Ren, Huanjie Wang, Yi Su, Ruiming Tang, Kun Gai, and Guorui Zhou. 2025. OneRec-Think: In-Text Reasoning for Generative Recommendation. *arXiv:2510.11639 [cs.IR]* <https://arxiv.org/abs/2510.11639>
 - [21] Xinchun Luo, Jiangxia Cao, Tianyu Sun, Jinkai Yu, Rui Huang, Wei Yuan, Hezheng Lin, Yichen Zheng, Shiyao Wang, Qigen Hu, et al. 2025. Qarm: Quantitative alignment multi-modal recommendation at kuaishou. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*. 5915–5922.
 - [22] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (London, United Kingdom) (KDD '18). Association for Computing Machinery, New York, NY, USA, 1930–1939. doi:10.1145/3219819.3220007
 - [23] Xu Ma, Pengjie Wang, Hui Zhao, Shaoguo Liu, Chuhan Zhao, Wei Lin, Kuang-Chih Lee, Jian Xu, and Bo Zheng. 2021. Towards a Better Tradeoff between Effectiveness and Efficiency in Pre-Ranking: A Learnable Feature Selection based Approach. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) (SIGIR '21). Association for Computing Machinery, New York, NY, USA, 2036–2040. doi:10.1145/3404835.3462979
 - [24] Frederic Morin and Yoshua Bengio. 2005. Hierarchical Probabilistic Neural Network Language Model. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. R5)*, Robert G. Cowell and Zoubin Ghahramani (Eds.). PMLR, 246–252. Reissued by PMLR on 30 March 2021.. <https://proceedings.mlr.press/r5/morin05a.html>
 - [25] Changhua Pei, Yi Zhang, Yongfeng Zhang, Fei Sun, Xiao Lin, Hanxiao Sun, Jian Wu, Peng Jiang, Junfeng Ge, Wenwu Ou, and Dan Pei. 2019. Personalized re-ranking for recommendation. In *Proceedings of the 13th ACM Conference on Recommender Systems* (Copenhagen, Denmark) (RecSys '19). Association for Computing Machinery, New York, NY, USA, 3–11. doi:10.1145/3298689.3347000
 - [26] Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan H. Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Q. Tran, Jonah Samost, Maciej Kula, Ed H. Chi, and Maheswaran Sathiamoorthy. 2023. Recommender Systems with Generative Retrieval. *arXiv:2305.05065 [cs.IR]* <https://arxiv.org/abs/2305.05065>
 - [27] Jinbo Song, Ruoran Huang, Xinyang Wang, Wei Huang, Qian Yu, Mingming Chen, Yafei Yao, Chaosheng Fan, Changping Peng, Zhangang Lin, Jinghe Hu, and Jingping Shao. 2022. Rethinking Large-scale Pre-ranking System: Entire-chain Cross-domain Models. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (Atlanta, GA, USA) (CIKM '22). Association for Computing Machinery, New York, NY, USA, 4495–4499. doi:10.1145/3511808.3557683
 - [28] Otilia Stretcu, Emmanouil Antonios Platanios, Tom M. Mitchell, and Barnabás Póczos. 2021. Coarse-to-Fine Curriculum Learning. *arXiv preprint arXiv:2106.04072* (2021).
 - [29] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential Recommendation with Bidirectional Encoder Representations from Transformer. *arXiv:1904.06690 [cs.IR]* <https://arxiv.org/abs/1904.06690>
 - [30] Juntao Tan, Shuyuan Xu, Wenyue Hua, Yingqiang Ge, Zelong Li, and Yongfeng Zhang. 2024. IDGenRec: LLM-RecSys Alignment with Textual ID Learning. *arXiv:2403.19021 [cs.IR]* <https://arxiv.org/abs/2403.19021>
 - [31] Yubao Tang, Ruqing Zhang, Jiafeng Guo, and Maarten de Rijke. 2023. Recent Advances in Generative Information Retrieval. In *SIGIR-AP 2023: 1st International ACM SIGIR Conference on Information Retrieval in the Asia Pacific*. ACM, 294–297.
 - [32] Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022. Transformer memory as a differentiable search index. *Advances in neural information processing systems* 35 (2022), 21831–21843.
 - [33] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).
 - [34] Wenjie Wang, Honghui Bao, Xinyu Lin, Jizhi Zhang, Yongqi Li, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. 2024. Learnable Item Tokenization for Generative Recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management* (Boise, ID, USA) (CIKM '24). Association for Computing Machinery, New York, NY, USA, 2400–2409. doi:10.1145/3627673.3679569
 - [35] Yujing Wang, Yingyan Hou, Haonan Wang, Ziming Miao, Shibin Wu, Qi Chen, Yuqing Xia, Chengmin Chi, Guoshuai Zhao, Zheng Liu, et al. 2022. A neural corpus indexer for document retrieval. *Advances in Neural Information Processing Systems* 35 (2022), 25600–25614.
 - [36] Ye Wang, Jiahao Xun, Minjie Hong, Jieming Zhu, Tao Jin, Wang Lin, Haoyuan Li, Linjun Li, Yan Xia, Zhou Zhao, and Zhenhua Dong. 2024. EAGER: Two-Stream Generative Recommender with Behavior-Semantic Collaboration. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Barcelona, Spain) (KDD '24). Association for Computing Machinery, New York, NY, USA, 3245–3254. doi:10.1145/3637528.3671775
 - [37] Zhe Wang, Liqin Zhao, Biye Jiang, Guorui Zhou, Xiaoqiang Zhu, and Kun Gai. 2020. Cold: Towards the next generation of pre-ranking system. *arXiv preprint arXiv:2007.16122* (2020).
 - [38] Yinwei Wei, Xiang Wang, Qi Li, Liqiang Nie, Yan Li, Xuanping Li, and Tat-Seng Chua. 2021. Contrastive learning for cold-start recommendation. In *Proceedings of the 29th ACM international conference on multimedia*. 5382–5390.
 - [39] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. 2021. SoundStream: An End-to-End Neural Audio Codec. *arXiv:2107.03312 [cs.SD]* <https://arxiv.org/abs/2107.03312>
 - [40] Jiaqi Zhai, Lucy Liao, Xing Liu, Yueming Wang, Rui Li, Xuan Cao, Leon Gao, Zhaojie Gong, Fangda Gu, Michael He, Yinghai Lu, and Yu Shi. 2024. Actions Speak Louder than Words: Trillion-Parameter Sequential Transducers for Generative Recommendations. *arXiv:2402.17152 [cs.LG]* <https://arxiv.org/abs/2402.17152>
 - [41] 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 (ICDE)*. IEEE, 1435–1448.