SAME TARGET, DIFFERENT PATHS: LEARNING FROM MULTIPLE REASONING TRAJECTORIES FOR GENERATIVE RECOMMENDATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning-based generative recommenders can support both explicit reasoning and direct SID generation within the same model. We study how multi-trajectory reasoning post-training affects this hybrid inference setting. In self-sampled rejection-sampling fine-tuning (RFT), multiple successful trajectories for the same request provide different reasoning paths while terminating at a shared target SID, coupling richer reasoning supervision with repeated terminal-target supervision. We introduce target-masked RFT (TM-RFT), which removes the terminal-target loss from augmentation trajectories while preserving supervision on their reasoning traces and on ordinary SFT data. On the OneReason benchmark, TM-RFT with up to four trajectories per request achieves a Weighted Pass@64 of 0.7576, compared with 0.6912 for SFT-only and 0.7011 for a group-normalized control that rescales the repeated target loss. Answer-supervised variants achieve higher standalone reasoning and direct scores, while showing much stronger agreement between the two inference modes, with up to 44% of their candidates shared. TM-RFT maintains route separation close to SFT-only and retains 99.6% of its strongest single-mode performance under a fixed 64-candidate budget. On three public Amazon datasets, TM-RFT improves the two-route hit rate over both the checkpoints and SFT-only.

1. INTRODUCTION

Generative recommendation formulates item prediction as the generation of item identifiers, often represented as hierarchical semantic IDs (SIDs) (Rajput et al., 2023). LLM-based formulations extend this paradigm by incorporating explicit reasoning into the generation process, allowing the same model to either reason before predicting the target SID or generate the SID directly (Hong et al., 2025).

The benefit of explicit reasoning varies across tasks. Large-scale evaluations of chain-of-thought prompting find its clearest gains in mathematical and symbolic reasoning, with much smaller improvements on many other tasks (Sprague et al., 2025). Recent work has therefore increasingly explored models that support both thinking and non-thinking inference. Qwen3 integrates the two modes within a unified model (Yang et al., 2025), while large hybrid-reasoning models further study how a model can adaptively choose when to engage in explicit reasoning (Jiang et al., 2025).

This distinction is also relevant to generative recommendation. Zhang et al. (2026) report that introducing textual chain-of-thought reasoning into Semantic-ID-based foundation recommenders can reduce recommendation performance in some settings. OneReason similarly evaluates thinking and non-thinking as distinct inference modes, reports no clear advantage for thinking in its preliminary studies, and further strengthens reasoning through post-training (OneRec Team et al., 2026). Together, these findings motivate examining how reasoning-oriented post-training behaves when both thinking and direct decoding are supported by the same model.

One post-training approach is to learn from model-generated reasoning trajectories. Rejection-sampling fine-tuning (RFT) filters sampled trajectories by outcome and reuses the accepted ones as training examples (Zelikman et al., 2022; Yuan et al., 2023). Multiple successful trajectories for the same input expose the model to different ways of reaching a valid answer, providing a broader set of reasoning demonstrations than a single trajectory. In generative recommendation, these trajectories

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

may follow different reasoning paths while terminating at the same target SID, coupling richer reasoning supervision with repeated supervision on a shared terminal target.

We introduce **target-masked RFT (TM-RFT)** to separate these two aspects of multi-trajectory training. TM-RFT preserves supervision on the sampled reasoning traces while removing the loss on their shared terminal target. Specifically, we mask the final domain marker and three SID tokens on selected augmentation examples, while ordinary SFT examples retain full answer supervision.

Across the main benchmark, TM-RFT with up to four trajectories per request achieves a Weighted Pass@64 of 0.7576, improving over 0.6912 for SFT-only and 0.7011 for a group-normalized control that rescales the repeated target loss. To characterize performance when both inference modes are used, we measure *hybrid retention*: the fraction of the strongest single-route performance preserved after splitting a fixed candidate budget between reasoning and direct decoding. TM-RFT achieves 99.6% hybrid retention under the $32{+}32$ allocation, compared with 90.8% for the group-normalized control, whose two routes converge on largely the same candidates.

Our contributions are summarized as follows:

- We study multi-trajectory RFT for generative recommendation, focusing on how learning from multiple successful reasoning paths shapes thinking behavior when the same model supports both reasoning and direct decoding.
- We propose **target-masked RFT (TM-RFT)**, which preserves supervision on successful reasoning traces while masking their terminal-target loss. This separates learning from multiple reasoning paths from repeated target supervision on the corresponding augmentation examples.
- We evaluate TM-RFT under reasoning, direct, and mixed inference settings. TM-RFT achieves the strongest benchmark result among the compared training variants and retains 99.6% of its strongest single-mode performance under mixed inference.

2. Related Work

Reasoning in generative recommendation. TIGER establishes hierarchical semantic IDs as an autoregressive item representation for generative retrieval (Rajput et al., 2023). Building on LLM-based recommendation, GREAM combines collaborative-semantic alignment, reasoning curriculum training, and reinforcement-learning post-training (Hong et al., 2025). WhisperRec approaches the inference-cost problem from a different direction by distilling explicit reasoning into latent tokens (Jiang et al., 2026). OneReason develops a broader training recipe spanning recommendation-oriented pre-training, supervised fine-tuning, and reinforcement learning (OneRec Team et al., 2026). Our work concentrates on the supervised post-training stage and, in particular, on how self-generated reasoning data are used within that stage.

Self-generated reasoning has also been widely used in general reasoning post-training. STaR iteratively bootstraps from model-generated rationales associated with correct answers (Zelikman et al., 2022), while Yuan et al. (2023) study rejection-sampling fine-tuning at scale and show that training benefits from a larger and more diverse set of correct reasoning solutions. These works establish successful model generations as a useful source of post-training data. We study how the supervision attached to such data should be assigned in generative recommendation.

Selective supervision within reasoning trajectories. Prior work has explored assigning different training roles to different parts of a reasoning trajectory. Ye et al. (2025) distinguish reasoning tokens from repetitive boilerplate and place greater emphasis on the former. Masked Thought masks subsets of intermediate chain-of-thought tokens during training (Chen et al., 2024). Liu et al. (2026) show that CoT training in language-model agents can also strengthen direct action prediction from the prompt, and explore masking final-action supervision on selected examples. Our setting similarly distinguishes supervision on the reasoning trace from supervision on the terminal SID in self-sampled recommendation post-training.

3. TASK SETUP

Figure 1 gives a simplified view of the recommendation task considered in this work.

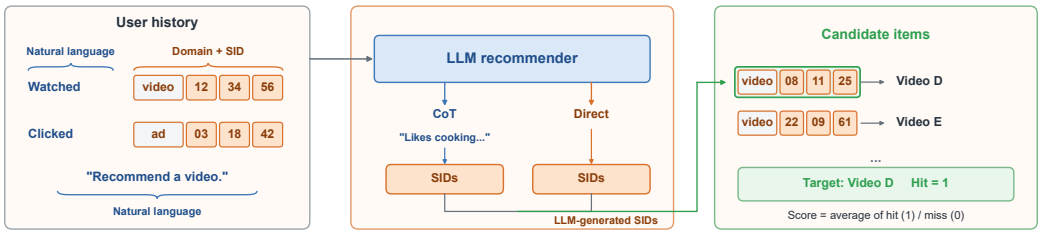

Figure 1: Illustration of generative recommendation setup. A user interaction history is given to an LLM recommender, which generates candidate semantic IDs (SIDs) either with an explicit reasoning trace or through direct decoding. A request is counted as a hit when at least one generated candidate matches a held-out target.

Input and output. The input to the model contains a user's past interactions from different recommendation domains together with natural-language prompts. The goal is to predict items that the user is likely to interact with next, based on their interaction history. Items are represented by semantic IDs (SIDs), consisting of a domain marker and hierarchical discrete tokens that encode semantic information about the item.

LLM recommendation. The recommender autoregressively generates SID tokens from the user history as context. In reasoning mode, it first generates a natural-language rationale (CoT) before producing the SID; in direct mode, it generates the SID directly.

Evaluation. For each request, the generated candidates are compared with the held-out target items. A request is counted as a *hit* if at least one target appears among the candidates. Pass@K is the average hit rate when the model generates K candidates. The main benchmark uses Pass@64, with separate evaluation for Video, Product, Ads, and Live.

4. TARGET-MASKED RFT

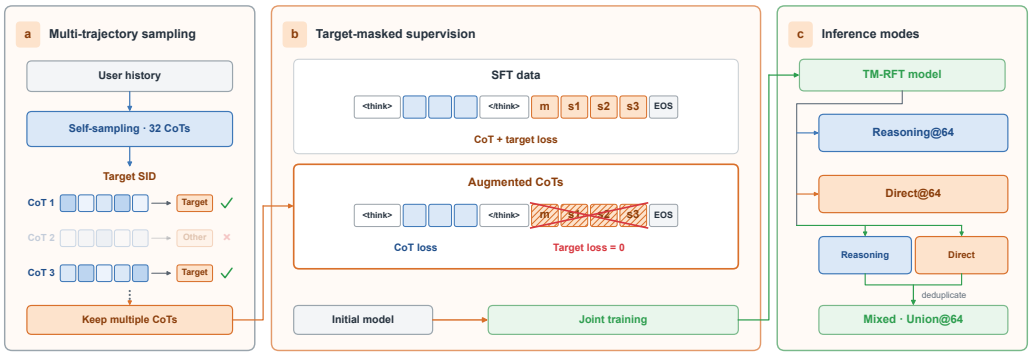

Figure 2: Overview of TM-RFT, covering multi-trajectory sampling, target-masked supervision, and reasoning/direct inference.

Figure 2 summarizes the full TM-RFT pipeline: successful reasoning trajectories are collected through self-sampling, their terminal-target loss is masked during augmentation training, and the resulting model is evaluated under reasoning, direct, and mixed inference.

Inference modes and hybrid retention. At inference time, the model can generate multiple candidate SIDs for each request. We refer to the number of generated candidates as the *candidate budget*. Candidates can be produced through reasoning, direct decoding, or both; when both modes are used, their candidate sets are deduplicated and merged before evaluation.

With a fixed budget of 64, let B_T candidates come from reasoning and B_D from direct decoding, with $B_T + B_D = 64$. We use *hybrid retention* to measure how much of a model's strongest single-mode performance is preserved when this budget is shared between the two modes:

$$R_{\text{hybrid}}(B_T, B_D) = \frac{S_{B_T,B_D}}{\max(S_{64,0}, S_{0,64})},$$

where S denotes Weighted Pass@64 under the corresponding allocation. A value close to 1 means that mixed inference retains most of the model's best single-mode performance.

Multi-trajectory RFT. For each request–target pair $g = (x_g, y_g)$, self-sampled RFT may retain n_g successful reasoning trajectories $\{c_{g1}, \ldots, c_{gn_g}\}$, all terminating at the same target y_g. The retained augmentation trajectories are generated in advance and kept fixed across the controlled training variants.

Each augmentation trajectory contains a reasoning trace and a terminal target. For trajectory (x_g, c_{gj}, y_g), we decompose the training loss into a reasoning component ℓ_P and a terminal-target component ℓ_Y, where ℓ_Y covers the final domain marker and three SID tokens:

$$\mathcal{L}_{\text{aug}} = \sum_{g} \sum_{j=1}^{n_g} \left[\ell_P(x_g, c_{gj}, y_g) + \alpha_{gj}\ell_Y(x_g, c_{gj}, y_g)\right].$$

Here, α_{gj} controls how much terminal-target loss is retained for trajectory j. TM-RFT sets $\alpha_{gj} = 0$ on the augmentation trajectories (Figure 2b). The reasoning traces remain supervised, while the final domain marker and SID tokens are excluded from the augmentation loss. Ordinary SFT examples continue to receive full supervision in the same fine-tuning run. Adding more accepted trajectories therefore increases reasoning supervision without adding terminal-target loss from those trajectories.

We compare TM-RFT with two variants that retain target supervision. Full-RFT uses $\alpha_{gj} = 1$ for every augmentation trajectory. The group-normalized variant uses $\alpha_{gj} = \frac{1}{n_g}$, giving each request-target group a total terminal-target weight of one, independent of the number of retained trajectories. The augmentation data, initialization, and optimization recipe are identical across the three variants.

5. Experimental Design

5.1. Main Experiments

Our main experiments use the publicly released OneReason-8B-Pretrain (Competition) checkpoint, an 8B recommendation foundation model pretrained to align itemic-token representations with natural language, user behavior, and recommendation signals (OneRec Team et al., 2026). Each item is represented by a domain-aware begin token (e.g., <|video_begin|>) followed by three hierarchical SID tokens, with a codebook size of 8,192 at each level.

We use full-parameter fine-tuning with AdamW, gradient clipping at 1.0, weight decay 0.01, and bf16 precision. The learning rate follows a cosine schedule with a peak value of 6×10^{-6} and a warmup ratio of 0.03. Training uses a global batch size of eight and runs for two epochs. Examples are packed into 32,768-token sequences with block-diagonal attention.

Ordinary SFT data and self-sampled augmentation trajectories are trained jointly. For comparisons among Full-RFT, group-normalized Full-RFT, and TM-RFT, we keep the initialization, augmentation data, and optimization recipe fixed and vary the supervision on augmentation trajectories according to Section 4.

5.2. Self-Sampled Augmentation

For the main experiments, we sample reasoning trajectories for 36,833 recommendation requests. Each request is sampled 32 times with temperature 0.8, top-p 0.95, and a maximum generation length of 1,500 tokens. A trajectory is accepted when its terminal SID exactly matches the target SID.

At least one accepted trajectory is obtained for 3,366 requests (9.14%). After exact-string deduplication of the reasoning traces, 7,775 trajectories remain, averaging 2.31 trajectories per accepted request. We cap the number of retained trajectories per request at $k \in \{1, 2, 4, 8\}$, yielding 3,366, 4,798, 6,151, and 7,175 augmentation trajectories, respectively. Unless otherwise specified, $k = 4$ is the main TM-RFT configuration.

5.3. OneReason Evaluation

Main benchmark. We evaluate on the OneReason Evaluation Benchmark[1], covering Video, Product, Ads, and Live recommendation (OneRec Team et al., 2026). The benchmark defines high-value targets separately for each category: Product uses clicks, Video uses watch time above the 75th percentile within the corresponding duration bucket, Ads uses conversion actions, and Live uses first gifting events. Its construction also includes abnormal-sample filtering, popularity downsampling, and category balancing.

The main inference setting allocates 32 candidates to reasoning and 32 to direct decoding. The two sets are deduplicated and merged, and we report the resulting category-level U@64 scores. The benchmark aggregate is

$$\text{Weighted Pass@64} = 9.6P_{\text{video}} + 3.4P_{\text{product}} + 1.4P_{\text{ads}} + 0.9P_{\text{live}},$$

where P_d denotes the corresponding Pass@64 on the $[0, 1]$ scale. The benchmark-defined weights are fixed across all methods.

Inference allocations. To examine performance when reasoning and direct decoding share the candidate budget, we additionally evaluate reasoning-only and direct-only decoding with all 64 candidates, together with mixed $48 + 16$, $32 + 32$, and $16 + 48$ allocations. The $32 + 32$ allocation remains our main setting. Hybrid retention, defined in Section 4, measures how much of the strongest single-route performance is retained under mixed inference.

Route-level analysis. For the main $32 + 32$ setting, we further separate the combined hit rate into requests retrieved only by reasoning, only by direct decoding, or by both routes. We also report candidate-set overlap and Top-1 agreement between the two routes on Video requests. These measurements characterize how the two inference modes contribute to the same combined candidate set.

5.4. Cross-Dataset Evaluation

We separately evaluate TM-RFT on three public Amazon recommendation datasets (Games, Office, and Industrial) using the train/test splits released by He et al. (2026). These experiments use the publicly released Evo-Rec Qwen3-1.7B checkpoints as their training starting point (Zhu et al., 2026).

For each training request, we sample eight reasoning trajectories and retain up to four successful, non-duplicate trajectories whose predicted SID matches the target. SFT-only continues training on one direct-answer example per request. TM-RFT uses the same SFT data and additionally trains on the retained reasoning trajectories with their answer-SID loss masked. Both variants start from the same Evo-Rec checkpoint and follow the same one-epoch training schedule.

We also evaluate the released SIDReasoner checkpoints from He et al. (2026) as an additional reference, without further training. Test requests are deduplicated by (history, target). At inference time, reasoning and direct decoding each use 32-beam SID-constrained search, and we report the hit rate of their deduplicated union, U(32+32), for each dataset and for the pooled test set.

[1] https://huggingface.co/datasets/OpenOneRec/OneReason-Eval-Benchmark

Table 1: Main results on the benchmark. The four middle columns report domain-level union U@64 (%); Weighted Pass@64 is the benchmark-defined aggregate score. k denotes the upper limit on successful trajectories per request. Best results in each column are shown in bold.

Method	Video U@64	Product U@64	Ads U@64	Live U@64	Weighted Pass@64
SFT-only	1.46	4.44	15.11	**20.94**	0.6912
TM-RFT ($k = 1$)	1.39	**5.22**	16.75	19.34	0.7198
TM-RFT ($k = 2$)	1.50	4.81	17.20	19.28	0.7218
TM-RFT ($k = 4$)	**1.59**	5.14	**17.53**	20.47	**0.7576**
TM-RFT ($k = 8$)	1.37	5.13	16.88	18.79	0.7117
Full-RFT	1.39	3.69	14.57	20.36	0.6460
Group-normalized Full-RFT	1.42	4.46	16.80	19.73	0.7011

Table 2: Weighted Pass@64 under different allocations of a fixed 64-candidate inference budget between reasoning and direct decoding (reasoning+direct). Hybrid retention is reported for the main $32 + 32$ setting and defined in Section 4. Best results in each column are shown in bold.

	Single route (64)		Reasoning + direct			
Method	Reasoning	Direct	48+16	32+32	16+48	Hybrid retention
SFT-only	0.6396	0.7131	0.6561	0.6912	0.7289	96.9%
TM-RFT ($k = 1$)	0.6883	0.7560	0.7043	0.7198	0.7635	95.2%
TM-RFT ($k = 2$)	0.6743	0.7510	0.6837	0.7218	**0.7683**	96.1%
TM-RFT ($k = 4$)	0.6954	0.7605	**0.7365**	**0.7576**	0.7668	**99.6%**
TM-RFT ($k = 8$)	0.6702	0.7493	0.6743	0.7117	0.7596	95.0%
Full-RFT	0.6642	0.7458	0.6400	0.6460	0.6755	86.6%
Group-normalized Full-RFT	**0.7081**	**0.7722**	0.6990	0.7011	0.7216	90.8%

6. Results and Analysis

Main results. Table 1 reports the benchmark performance across training variants and trajectory caps. TM-RFT with $k = 4$ achieves the strongest aggregate result, with a Weighted Pass@64 of 0.7576, and the highest Video and Ads scores; SFT-only remains strongest on Live and TM-RFT with $k = 1$ on Product. Both answer-supervised variants fall below SFT-only on the aggregate score: Full-RFT reaches 0.6460, and group-normalizing the repeated target loss recovers part of this loss (0.7011) but remains below every TM-RFT variant, indicating that target-loss rescaling alone does not account for the gain from masking.

All TM-RFT variants improve over SFT-only, and the trajectory cap affects the size of the gain: the aggregate score rises from $k = 1$ to $k = 4$ and declines at $k = 8$, which falls below $k = 1$. We therefore use $k = 4$ as the main TM-RFT configuration in the analyses below.

Performance retention under mixed inference. We next examine how the same checkpoints behave when the fixed 64-candidate budget is shared between reasoning and direct decoding. Under the main $32+32$ setting, TM-RFT with $k = 4$ retains 99.6% of its strongest single-route performance, the highest hybrid retention among the compared variants (Table 2).

We also evaluate unequal $48 + 16$ and $16 + 48$ allocations. TM-RFT achieves the best score under all three mixed allocations: $k = 4$ is best for $48 + 16$ and $32 + 32$, while $k = 2$ is best for $16 + 48$. By comparison, group-normalized Full-RFT achieves the strongest reasoning-only and direct-only scores but shows substantially lower performance retention once the candidate budget is shared between the two modes.

Route overlap and unique coverage. To understand why stronger single routes do not necessarily yield a stronger combined retriever, we split the two-route score H_U into hits retrieved only by the reasoning route, only by the direct route, and by both (Table 3). The three parts sum to H_U, so

Table 3: Route contribution on the OneReason benchmark (32 candidates per route). H_U is the two-route Weighted Pass@64 from Table 1, split into hits retrieved only by the reasoning route, only by the direct route, or by both (the three sum to H_U). Overlap is the mean share of candidates common to both routes, and Top-1 the fraction of requests whose routes rank the same item first (Video). Best results are shown in bold.

		Hits retrieved by			Route agreement	
Method	H_U	Reasoning only	Direct only	Both	Overlap	Top-1
SFT-only	0.6912	0.1773	0.1986	0.3152	12.7%	3.2%
TM-RFT ($k = 1$)	0.7198	0.1834	0.1756	0.3608	10.0%	1.5%
TM-RFT ($k = 2$)	0.7218	0.1856	0.1852	0.3509	10.8%	1.7%
TM-RFT ($k = 4$)	**0.7576**	**0.2015**	**0.2087**	0.3473	12.4%	2.3%
TM-RFT ($k = 8$)	0.7117	0.1817	0.1733	0.3568	11.8%	2.3%
Full-RFT	0.6460	0.1123	0.1631	0.3706	44.0%	34.6%
Group-normalized Full-RFT	0.7011	0.1381	0.1600	0.4029	43.6%	34.4%

Table 4: Two-route hit rate U(32+32) (%) on three Amazon datasets. *SIDReasoner* and *Evo-Rec* are the released checkpoints (Qwen3-1.7B) without further training; *SFT-only* and *TM-RFT* continue the Evo-Rec checkpoint for one epoch on the same data, differing only in whether augmentation rows are added with their answer-SID loss masked. Test requests are deduplicated by (history, target). Best result in each row is shown in bold.

Dataset	#Test	SIDReasoner	Evo-Rec	SFT-only	TM-RFT
Amazon Games	6,018	20.01	22.42	**22.85**	22.30
Amazon Office	4,828	23.94	24.75	24.79	**26.37**
Amazon Industrial	4,485	22.83	24.62	24.06	**25.53**
Pooled	15,331	22.07	23.79	23.81	**24.53**

the combined score is determined by the coverage contributed by each route and their shared hits. All quantities are weighted with the benchmark weights and are therefore directly comparable with Table 1.

Table 3 shows a clear difference in route agreement across supervision schemes. Full-RFT and group-normalized Full-RFT produce substantially higher agreement, with candidate overlap around 44% and Top-1 agreement around 34%. Under TM-RFT, candidate overlap remains between 10.0% and 12.4%, while Top-1 agreement stays between 1.5% and 2.3%, close to the SFT-only level. Retaining terminal-target supervision on the augmentation trajectories is therefore associated with much stronger agreement between the reasoning and direct routes.

This difference is also reflected in how the two routes contribute to the combined score. Group-normalized Full-RFT achieves the strongest standalone scores, but many of its hits are common to both decoding modes. TM-RFT with $k = 4$ instead has the largest reasoning-only and direct-only contributions, 0.2015 and 0.2087, yielding the highest combined score of 0.7576. The drop at $k = 8$ is accompanied by smaller unique contributions from both routes, while route agreement remains low.

Cross-dataset evaluation. To test whether TM-RFT remains effective on top of an existing reasoning recommender, we start from the publicly released Evo-Rec Qwen3-1.7B checkpoints (Zhu et al., 2026), which have already undergone the full Evo-Rec training pipeline, including reinforcement learning. We evaluate on three Amazon review categories (Video Games, Office Products, and Industrial and Scientific) using the train/test splits released by He et al. (2026). For each training request, we sample eight reasoning trajectories and retain up to four successful, non-duplicate trajectories whose predicted SID matches the target. As shown in Table 4, *SFT-only* continues training on one direct-answer example per request, whereas *TM-RFT* additionally trains on the retained trajectories with the answer-SID loss masked. Both variants start from the same checkpoint and use the same one-epoch training schedule. As a further reference, we evaluate the released SIDReasoner

checkpoints (He et al., 2026), which are also Qwen3-1.7B models trained with reinforcement learning on the same splits, without further training. At inference, we combine candidates from a reasoning route and a direct route, each decoded with 32-beam SID-constrained search, and report the hit rate of their deduplicated union, U(32+32).

7. Discussion and Limitations

Concurrent work. A concurrent preprint, Evo-Rec (Zhu et al., 2026), released during the finalization of our manuscript, also studies reasoning trajectories for Semantic-ID-based generative recommendation. The two works address complementary questions. Evo-Rec focuses on selecting and optimizing effective reasoning trajectories, retaining a single highest-utility trajectory per instance during supervised training. In contrast, we study what should be supervised when multiple successful trajectories are retained for the same request. TM-RFT removes repeated terminal-target supervision from these augmentation trajectories while preserving their reasoning signals. Thus, Evo-Rec determines which trajectories to learn from, whereas TM-RFT determines which tokens of the retained trajectories to supervise. The cross-dataset evaluation in Section 6 combines the two: continuing the released Evo-Rec checkpoints with TM-RFT improves their two-route hit rate on the pooled Amazon test set.

Multi-trajectory supervision and two-route retrieval. The results highlight a distinction between improving an individual decoding route and improving the model under two-route inference. Group-normalized Full-RFT achieves the strongest reasoning-only and direct-only scores, yet TM-RFT performs better when the same inference budget is divided between the two modes, under every split we tested. The route decomposition in Table 3 locates the difference: answer supervision on the augmentation trajectories makes the two routes retrieve largely the same candidates, so their union adds little; whereas TM-RFT keeps them as separate as SFT-only while raising the unique contribution of each. TM-RFT preserves the learning signal from the reasoning paths without repeatedly applying terminal-target loss to every augmented trajectory.

Sample selection and token supervision. Success filtering and target masking affect different parts of the training process. Requests that yield more successful trajectories contribute more augmentation examples and therefore receive greater exposure. Group-level normalization changes the relative weight contributed by those examples, whereas TM-RFT leaves the selected trajectories and their rationale losses in place while removing their terminal-target loss.

This distinction is reflected in the controls used in our experiments. Group normalization reduces the repeated target weight associated with multiple successful trajectories from the same request, while TM-RFT removes that repeated target supervision from the augmentation data. Both operate after successful trajectories have been selected, so the composition of the augmentation set itself remains unchanged.

The role of answer supervision. TM-RFT masks the terminal target only on successful augmentation trajectories, while ordinary SFT examples retain full answer supervision. The effect of this masking therefore depends on the supervision already provided by the base SFT data. On the OneReason benchmark, the SFT data already includes reasoning-format examples with supervised targets, and masking all augmentation targets gives the best result. In the Amazon setting, however, the ordinary SFT data contains only direct-answer examples. Fully masked TM-RFT therefore provides no target supervision after a reasoning trace. It still improves over SFT-only and the released checkpoint, but underperforms Full-RFT and group-normalized Full-RFT on the two-route hit rate and shows higher overlap between reasoning and direct decoding. Supervising the target on one trajectory per request, while masking the remaining trajectories, closes this gap and returns the overlap to the answer-supervised level. These results suggest that target masking works best when reasoning-conditioned answer supervision is already present elsewhere in training; TM-RFT then removes the repeated target supervision introduced by additional trajectories.

Limitations. First, we evaluate TM-RFT on two scales within the Qwen3 family: OneReason-8B on the OneReason benchmark and Qwen3-1.7B on three Amazon review categories. Whether

the observed effects generalize to other model architectures, scales, and recommendation domains remains to be studied.

Second, TM-RFT is a supervised post-training method. In experiments initialized from the released Evo-Rec checkpoints (Zhu et al., 2026), we apply TM-RFT after their reinforcement-learning stage, without subsequent RL or on-policy distillation (OPD) (Agarwal et al., 2024). It remains unclear how TM-RFT interacts with these later-stage optimization methods, or whether OPSD and related techniques can reduce the sampling cost of constructing the augmentation pool (Zhao et al., 2026); on OneReason, we currently sample 32 trajectories per request.

Third, all evaluations are offline, and TM-RFT has not yet been validated in a production recommender. Our augmentation pool is generated once and kept fixed, whereas production systems are typically retrained periodically on fresh interaction logs. Efficiently integrating TM-RFT into such recurring training pipelines, and determining whether its offline gains translate into online improvements under A/B testing, remain important directions for future work.

8. Conclusion

We studied multi-trajectory reasoning post-training for generative recommendation, focusing on the repeated terminal-target supervision introduced when multiple successful CoTs for the same request are added to training. TM-RFT retains the loss on these reasoning traces while masking their shared terminal target.

On the OneReason benchmark, TM-RFT with $k = 4$ achieves the strongest Weighted Pass@64 among the compared training variants. Its advantage is not explained by stronger isolated decoding routes: group-normalized Full-RFT performs better under both reasoning-only and direct-only decoding, while TM-RFT preserves substantially more of its single-route performance when the inference budget is split across the two modes. Across additional Amazon recommendation settings, TM-RFT also achieves the highest reasoning-route HR@10 among the compared variants.

Together, these results show that learning from multiple successful reasoning trajectories depends on how supervision is distributed between their diverse reasoning traces and their shared terminal target.

Reproducibility Statement

The supplementary material documents training configurations, data composition, trajectory-sampling procedures, token-level supervision rules, and decoding and evaluation protocols. It also describes route recombination and query-level statistical analysis, and includes machine-readable aggregates and lightweight validation scripts. The underlying private training data and full candidate dumps are not included, which limits independent end-to-end reproduction of the reported experiments.

Ethics Statement

Recommendation histories may contain sensitive behavioral information. Our conclusions concern offline candidate coverage and do not establish effects on user engagement, welfare, or safety. The supplementary materials provide aggregate results without releasing the underlying private training data or full candidate dumps. Any release of additional assets must comply with applicable data-use permissions and institutional requirements, protect user privacy, and avoid exposing identifiable behavioral histories. Future deployment studies should evaluate downstream effects beyond retrieval accuracy.

Statement of AI Use

Generative AI tools assisted with literature organization, manuscript drafting and translation, problem formulation, algebraic analysis, experimental design, and code for checking aggregate results. The use of model-generated training trajectories is described separately in the experimental protocol. The authors are responsible for reviewing AI-assisted material and for the correctness of the references, mathematical arguments, experimental configurations, code, empirical results, and interpretations presented in the final manuscript.

REFERENCES

Rishabh Agarwal, Nino Vieillard, Yongchao Zhou, Piotr Stanczyk, Sabela Ramos Garea, Matthieu Geist, and Olivier Bachem. On-policy distillation of language models: Learning from self-generated mistakes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/5be69a584901a26c521c2b51e40a4c20-Abstract-Conference.html.

Changyu Chen, Xiting Wang, Ting-En Lin, Ang Lv, Yuchuan Wu, Xin Gao, Ji-Rong Wen, Rui Yan, and Yongbin Li. Masked thought: Simply masking partial reasoning steps can improve mathematical reasoning learning of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5872–5900, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.320. URL https://aclanthology.org/2024.acl-long.320/.

Yingzhi He, Yan Sun, Junfei Tan, Yuxin Chen, Xiaoyu Kong, Chunxu Shen, Xiang Wang, An Zhang, and Tat-Seng Chua. Reasoning over semantic IDs enhances generative recommendation, 2026. URL https://arxiv.org/abs/2603.23183.

Minjie Hong, Zetong Zhou, Zirun Guo, Ziang Zhang, Ruofan Hu, Weinan Gan, Jieming Zhu, and Zhou Zhao. Generative reasoning recommendation via LLMs, 2025. URL https://arxiv.org/abs/2510.20815.

Hao Jiang, Peiru Du, Pengfei Yao, Mengting Li, Siyuan Lou, Kuo Cai, Sheng Yu, Qiang Luo, Jian Liang, Ruiming Tang, Fei Pan, Peng Jiang, and Wenwu Ou. WhisperRec: Latent reasoning for efficient foundation recommendation models, 2026. URL https://arxiv.org/abs/2607.26621.

Lingjie Jiang, Xun Wu, Shaohan Huang, Qingxiu Dong, Zewen Chi, Li Dong, Xingxing Zhang, Tengchao Lv, Lei Cui, and Furu Wei. Think only when you need with large hybrid-reasoning models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 176843–176870, 2025. doi: 10.52202/085713-5323. URL https://papers.neurips.cc/paper_files/paper/2025/hash/e9a32fade47b906de908431991440f7c-Abstract-Conference.html.

Jingyu Liu, Zhiwen Wang, Yuxin Jing, Huanyu Zhou, and Yong Liu. Where do CoT training gains land in LLM based agents?, 2026. URL https://arxiv.org/abs/2606.26935.

OneRec Team, Biao Yang, Boyang Ding, Chenglong Chu, Dunju Zang, Fei Pan, Han Li, Hao Jiang, Honghui Bao, Huanjie Wang, Jian Liang, Jiangxia Cao, Jiao Ou, Jiaxin Deng, Jinghao Zhang, Kun Gai, Lu Ren, Peiru Du, Pengfei Zheng, Rongzhou Zhang, Ruiming Tang, Shiyao Wang, Siyang Mao, Siyuan Lou, Teng Shi, Wei Yuan, Wenlong Xu, Xingchen Liu, Xingmei Wang, Xinqi Jin, Yan Sun, Yan Wang, Yifei Hu, Yingzhi He, Yufei Ye, Yuhao Wang, Yunhao Zhou, Yuqin Dai, Zhao Liu, Zhipeng Wei, Zhixin Ling, Ziming Li, Zixing Zhang, Ziyuan Liu, An Zhang, Changxin Lao, Chaoyi Ma, Chengru Song, Defu Lian, Fan Yang, Guowang Zhang, Hao Peng, Jiayao Shen, Jie Chen, Jun Xu, Junmin Chen, Kun Zhang, Kuo Cai, Mingxing Wen, Minmao Wang, Minxuan Lv, Qi Zhang, Qiang Luo, Sheng Yu, Shijie Li, Shijie Yi, Shuang Yang, Shugui Liu, Shuni Chen, Tinghai Zhang, Tingting Gao, Xiang Wang, Xiangyu Wu, Xiangyu Zhao, Xiao Lv, Xiaoyou Zhou, Xuming Wang, Yong Du, Zejian Zhang, Zhaojie Liu, Zhiyang Zhang, Zhuang Zhuang, Ziqi Wang, and Ziyi Zhao. Onereason technical report, 2026. URL https://arxiv.org/abs/2606.06260.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Q. Tran, Jonah Samost, Maciej Kula, Ed H. Chi, and Maheswaran Sathiamoorthy. Recommender systems with generative retrieval. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-0452. URL https://proceedings.neurips.cc/paper/2023/hash/20dcab0f14046a5c6b02b61da9f13229-Abstract.html.

Zayne Sprague, Fangcong Yin, Juan Diego Rodriguez, Dongwei Jiang, Manya Wadhwa, Prasann Singhal, Xinyu Zhao, Xi Ye, Kyle Mahowald, and Greg Durrett. To CoT or not to CoT? chain-of-thought helps mainly on math and symbolic reasoning. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=w6nlcS8Kkn.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu,

Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Ziang Ye, Zhenru Zhang, Yang Zhang, Jianxin Ma, Junyang Lin, and Fuli Feng. Disentangling reasoning tokens and boilerplate tokens for language model fine-tuning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20939–20957, Vienna, Austria, July 2025. Association for Computational Linguistics. doi: 10.18653/v1/2025.findings-acl.1078. URL https://aclanthology.org/2025.findings-acl.1078/.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. Scaling relationship on learning mathematical reasoning with large language models, 2023. URL https://arxiv.org/abs/2308.01825.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 15476–15488, 2022. doi: 10.52202/068431-1126. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/639a9a172c044fbb64175b5fad42e9a5-Abstract-Conference.html.

Luankang Zhang, Yonghao Huang, Hang Lv, Mingjia Yin, Liangyue Li, Zulong Chen, Hao Wang, and Enhong Chen. Why thinking hurts? diagnosing and rectifying the reasoning shift in foundation recommender models, 2026. URL https://arxiv.org/abs/2602.16587.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=Jpxfof0EaS.

Mengdan Zhu, Yufan Zhao, Sophie Di, Yao Zhao, Tao Di, Yulan Yan, Sridhar Iyer, and Liang Zhao. Learning better reasoning for generative recommendation with semantic IDs, 2026. URL https://arxiv.org/abs/2609.29973.