HISTORY-CONDITIONED JOINT-PREFIX ALIGNMENT FOR GENERATIVE RECOMMENDATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Generative recommendation retrieves items by autoregressively generating semantic identifiers, but beam search may discard a target before its complete identifier is generated. Our preliminary analysis across three benchmarks shows that most missed targets are pruned within the first two decoding steps, highlighting the importance of early prefix retention. However, retaining a target's first-token branch alone is insufficient if its continuation is pruned at the next step: aligning only the first-token distribution improves first-token survival but yields little gain in complete-path retention. This observation motivates joint supervision of early branches and their continuations. We propose **P**refix **A**lignment with **T**emporal **H**istory (PATH), which aggregates transition statistics from the training corpus over recent interactions with exponential decay to construct history-conditioned two-token prefix targets. PATH aligns the model's joint predictions with these targets through forward KL divergence, using a chain-rule decomposition that enables unbiased Monte Carlo estimation. At inference time, PATH reuses the transition statistics for pointwise mutual information (PMI) calibration, reranking completed candidates relative to global prefix frequency. Experiments on Beauty, Instruments, and Yelp demonstrate the effectiveness of PATH, showing consistent improvements in recommendation performance and higher full-SID survival rates.

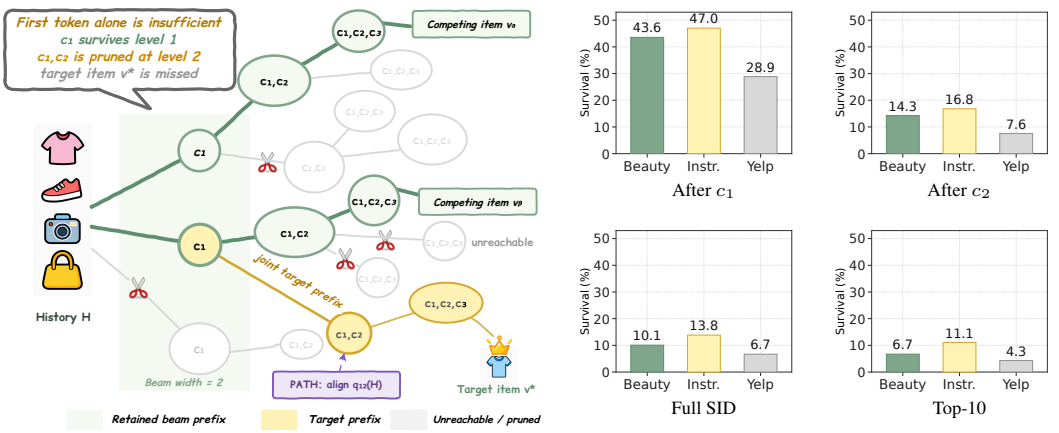
1 INTRODUCTION

Sequential recommendation aims to predict the next item a user will interact with from their interaction history (Yang et al., 2023; Dang et al., 2026; Wang et al., 2025). A generative approach to this task represents items as hierarchical semantic identifiers (SIDs) and predicts the next item by autoregressively generating its identifier (Rajput et al., 2023; Hou et al., 2025). To retrieve multiple candidate items, beam search extends identifier prefixes token by token, retaining only a limited set of high-scoring prefixes at each decoding step (Oshima et al., 2025; Wang & Han, 2025). Once a prefix is discarded, all items whose identifiers extend that prefix are excluded from further consideration (Figure 1(a)). A target item can therefore enter the final candidate set only if its path survives both early branch selection and subsequent prefix expansion.

Recent work has improved SID construction and its integration with recommendation training (Wang et al., 2024; Zheng et al., 2024; Liu et al., 2025; Fu et al., 2026). Prefix retention has also become an explicit optimization target. APAO adaptively emphasizes vulnerable prefixes (Yu et al., 2026), while BEAR encourages positive code tokens to remain within the Top-B set (Yang et al., 2026). These approaches primarily strengthen prefixes along the observed target path. We study how corpus-level transition statistics can complement this supervision by guiding predictions toward multiple plausible successor prefixes for the current history.

Our preliminary analysis reveals that improved first-token survival does not necessarily translate into better complete-path retention. Figure 1(b) shows a sharp decline in ground-truth survival within the first two decoding steps at beam width $B = 20$ across Beauty, Instruments, and Yelp. Targets pruned at these steps account for 91.9%–96.6% of all Hit@10 misses (Section 4.3). This concentration motivates supervision at early decoding steps, but the scope of that supervision matters. In our first-token-only alignment diagnostic, first-token survival increases on all three datasets, while second-prefix and full-SID survival remain close to those of the next-token prediction base-

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.



(a) Early Prefix Pruning **(b)** Ground-Truth Survival

Figure 1: Early beam pruning and ground-truth path survival. (a) A target can survive at c_1^* yet be irreversibly lost when c_2^* is removed at the next beam step. Through joint-prefix alignment, PATH aims to retain relevant target paths that would otherwise be pruned, while allowing irrelevant branches to be discarded. (b) Ground-truth prefix survival after c_1 and c_2, full-SID survival, and Hit@10 for the NTP baseline with $B = 20$ on Beauty, Instruments, and Yelp. Even when the first-token prefix survives, many target paths are lost at the second decoding step. The subsequent decline from two-token prefix survival to full-SID survival is smaller in absolute terms. This pattern motivates joint supervision of (c_1, c_2) to preserve both early branches and their continuations. All four plots report percentages on the same vertical scale.

line. These observations motivate supervision over early prefix combinations that couples first-token (c_1) branches with their second-token (c_2) continuations.

Historical transitions provide supervision over these early prefix combinations. Each next-token prediction (NTP) training instance directly supervises one observed successor, while transitions across the training corpus provide evidence about multiple plausible successors. We introduce Prefix Alignment with Temporal History (PATH) to incorporate this evidence into autoregressive training. PATH estimates transition distributions between two-token SID prefixes and aggregates the successor distributions associated with recent interactions using exponentially decaying weights. The resulting joint soft target over (c_1, c_2) retains the association between each first-token branch and its second-token continuations. PATH aligns the model's joint prediction with this history-conditioned target through forward KL divergence, complementing NTP supervision along the observed target path. For efficient computation, we decompose the joint KL by the chain rule and estimate its conditional term with Monte Carlo sampling. This yields unbiased estimates of both the auxiliary objective and its gradient without evaluating second-token predictions for every first-token branch.

The same transition statistics also provide a supplementary ranking signal after candidate generation. Raw transition probabilities reflect both history-specific associations and global prefix frequency. We use pointwise mutual information (PMI) calibration to measure association relative to a prefix's overall prevalence and combine the resulting signal with the model score. This step reorders the completed beam without changing which candidates survive.

Our contributions are as follows:

- We show that improving first-token survival alone yields little gain in full-SID survival across three datasets, motivating supervision of early prefix continuations.
- We propose PATH, which aligns early predictions with history-conditioned joint-prefix targets through unbiased stochastic KL estimation and supplements candidate ranking with PMI calibration.
- Through empirical evaluation, we demonstrate PATH's effectiveness in preserving target paths during decoding and improving recommendation performance across three datasets.

2 PRELIMINARIES

2.1 SEQUENTIAL RECOMMENDATION WITH SEMANTIC IDENTIFIERS

Given a user's interaction history $\mathcal{H} = (v_1, \ldots, v_n)$ over an item set $\mathcal{V}$, sequential recommendation predicts the next item $v^* = v_{n+1}$. A fixed tokenizer maps each item v to a length-L SID $c^v = (c_1^v, \ldots, c_L^v)$, where $L \geq 2$ and $c_l^v \in C_l$. We assume complete SIDs uniquely identify items. Let $c_{1:l}^v$ denote a length-l prefix and $c_{12}^v = (c_1^v, c_2^v)$ the joint prefix.

Let $X = (c^{v_1}, \ldots, c^{v_n})$ and $c^* = c^{v^*}$. The autoregressive model and its per-instance next-token prediction (NTP) loss are

$$p_\theta(c^* \mid X) = \prod_{l=1}^{L} p_\theta(c_l^* \mid X, c_{<l}^*), \tag{1}$$

$$\mathcal{L}_{\text{NTP}} = -\sum_{l=1}^{L} \log p_\theta(c_l^* \mid X, c_{<l}^*), \tag{2}$$

where $c_{<l}^*$ contains the preceding target tokens.

2.2 TRIE-CONSTRAINED DECODING AND PREFIX SURVIVAL

Trie-constrained beam search expands prefixes through valid trie edges and retains up to B highest-scoring extensions at each step. Let $\mathcal{B}_l(X)$ denote the retained length-l prefixes and $\mathcal{B}(X) \subseteq \mathcal{V}$ the completed item candidates. Once a target prefix is pruned, subsequent expansions cannot recover the target.

For an evaluation set $\mathcal{D} = \{(X_i, v_i^*)\}_{i=1}^M$, the level-$l$ prefix survival rate is

$$S_{c_l} = \frac{1}{M} \sum_{i=1}^{M} \mathbf{1}\left[c_{1:l}^{v_i^*} \in \mathcal{B}_l(X_i)\right], \tag{3}$$

where $\mathbf{1}[\cdot]$ is the indicator function. Full-SID survival is $S_{\text{full}} = S_{c_L}$, and irreversible pruning implies

$$1 \geq S_{c_1} \geq S_{c_2} \geq \cdots \geq S_{c_L} = S_{\text{full}}. \tag{4}$$

Full-SID survival measures candidate inclusion, whereas Hit@K additionally depends on ranking, with $\text{Hit@}K \leq S_{\text{full}}$. Reranking preserves the candidate set and all survival rates.

3 PATH: JOINT-PREFIX ALIGNMENT AND PMI CALIBRATION

PATH combines history-conditioned joint-prefix alignment during training with PMI-based candidate calibration during inference (Figure 2).

3.1 JOINT-PREFIX TRANSITION STATISTICS

This section uses two-token prefixes to retain these branch–continuation relationships while pooling interactions among items sharing a prefix, targeting the positions where our analysis finds most early target pruning (section 4.3).

Define $\pi_{12}(v) = (c_1^v, c_2^v)$, and let $\mathcal{P}_{12} \subseteq C_1 \times C_2$ denote the valid two-token prefixes induced by the fixed SID index. Transition counts are estimated exclusively from training interactions. We use $a = (a_1, a_2)$ and $b = (b_1, b_2)$ to denote source and destination joint prefixes, respectively. For each adjacent item pair (v_t, v_{t+1}), we count a transition from $a = \pi_{12}(v_t)$ to $b = \pi_{12}(v_{t+1})$. Let $C(a, b)$ be its count across training sequences and $C_{\text{out}}(a) = \sum_{b' \in \mathcal{P}_{12}} C(a, b')$. The smoothed transition distribution is

$$T(b \mid a) = \frac{C(a, b) + \varepsilon}{C_{\text{out}}(a) + \varepsilon|\mathcal{P}_{12}|}, \tag{5}$$

where $\varepsilon > 0$ provides additive smoothing.

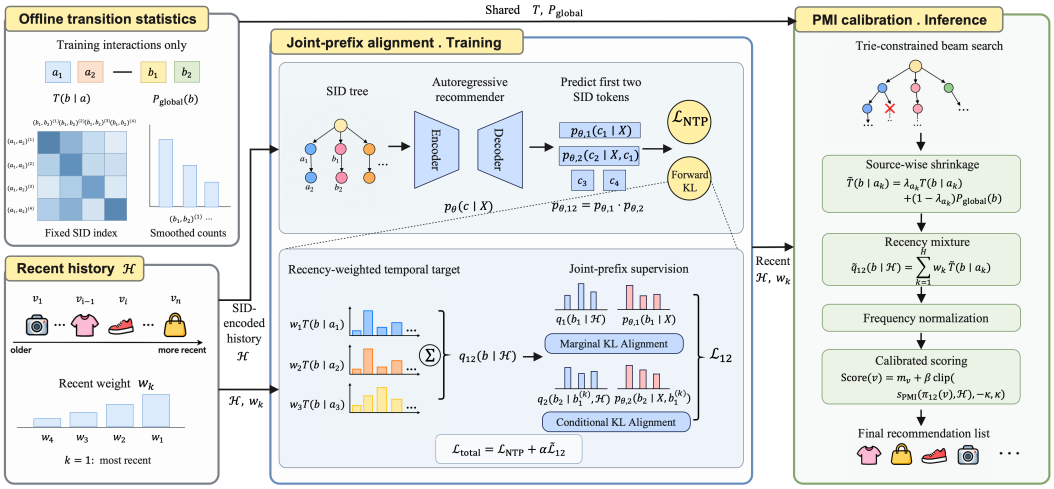


Figure 2: Overview of PATH. Joint-prefix transitions estimated from training interactions are aggregated over recent history to construct a temporal soft target. Joint forward-KL alignment supervises early decoding paths, while PMI-based calibration adjusts completed candidate scores relative to global prefix frequency.

We define the source-prefix distribution and the corresponding reference marginal as

$$P_{\text{src}}(a) = \frac{C_{\text{out}}(a)}{\sum_{a' \in \mathcal{P}_{12}} C_{\text{out}}(a')}, \quad P_{\text{global}}(b) = \sum_{a \in \mathcal{P}_{12}} P_{\text{src}}(a) T(b \mid a). \tag{6}$$

Thus, P_{global} describes destination-prefix frequency under the smoothed transition model.

3.2 Recency-Weighted Temporal Targets

Conditioning only on the last interaction makes the target sensitive to a single observation, while equal weighting fails to distinguish recent interactions from older ones. We aggregate a recent history window with exponential decay to retain several sources of transition evidence while emphasizing recency.

For a nonempty history $\mathcal{H} = (v_1, \ldots, v_n)$, let $H = \min(n, K_{\text{hist}})$. We construct the temporal soft target over destination joint prefixes by mixing the corresponding transition distributions

$$q_{12}(b \mid \mathcal{H}) = \sum_{k=1}^{H} w_k \, T(b \mid \pi_{12}(v_{n-k+1})), \tag{7}$$

where $w_k = \frac{\gamma^{k-1}}{\sum_{j=1}^{H} \gamma^{j-1}}$ is the normalized weight of the k-th most recent item, $0 < \gamma \leq 1, k \in \{1, \ldots, H\}$. For $\gamma < 1$, recent interactions receive greater weight, while earlier interactions within the window retain a nonzero contribution. When the history is empty, we set $q_{12}(b \mid \mathcal{H}) = P_{\text{global}}(b)$.

3.3 Joint Forward-KL Alignment

PATH aligns the joint two-token distribution to supervise both branch selection and within-branch continuations.

Given the SID-encoded history X, the model assigns probability to a destination prefix $b = (b_1, b_2)$ as $p_{\theta,12}(b_1, b_2 \mid X) = p_{\theta,1}(b_1 \mid X)\, p_{\theta,2}(b_2 \mid X, b_1)$, where $p_{\theta,1}$ and $p_{\theta,2}$ are the first- and second-token predictive distributions. Writing $q_{12}(b_1, b_2 \mid \mathcal{H})$ for $q_{12}((b_1, b_2) \mid \mathcal{H})$, we extend the target by zero outside $\mathcal{P}_{12}$ and define

$$q_1(b_1 \mid \mathcal{H}) = \sum_{b_2 \in C_2} q_{12}(b_1, b_2 \mid \mathcal{H}), \quad q_2(b_2 \mid b_1, \mathcal{H}) = \frac{q_{12}(b_1, b_2 \mid \mathcal{H})}{q_1(b_1 \mid \mathcal{H})}, \tag{8}$$

with the conditional defined for $q_1(b_1 \mid \mathcal{H}) > 0$. Below, we suppress conditioning on $\mathcal{H}$ and X.

By the chain rule, the joint forward KL decomposes as

$$\begin{aligned}\mathcal{L}_{12} &= \mathrm{KL}(q_{12}(\cdot \mid \mathcal{H}) \,\|\, p_{\theta,12}(\cdot \mid X)) \\ &= \mathrm{KL}(q_1(\cdot \mid \mathcal{H}) \,\|\, p_{\theta,1}(\cdot \mid X)) \\ &\quad + \mathbb{E}_{b_1 \sim q_1(\cdot \mid \mathcal{H})} \left[\mathrm{KL}(q_2(\cdot \mid b_1, \mathcal{H}) \,\|\, p_{\theta,2}(\cdot \mid X, b_1))\right].\end{aligned} \tag{9}$$

The first term aligns probabilities across first-token branches; the second aligns their continuations, weighted by the target branch probabilities. Forward KL penalizes insufficient model probability on prefix combinations carrying substantial target mass, encouraging coverage of empirically plausible paths.

The marginal term is computed exactly in the standard forward pass. Evaluating the conditional term exactly would require second-token predictions for every supported first-token branch. We draw K_{mc} independent samples $b_1^{(k)} \sim q_1$ with replacement and estimate

$$\begin{aligned}\widetilde{\mathcal{L}}_{12} &= \mathrm{KL}(q_1 \| p_{\theta,1}) \\ &\quad + \frac{1}{K_{\mathrm{mc}}} \sum_{k=1}^{K_{\mathrm{mc}}} \mathrm{KL}\Big(q_2(\cdot \mid b_1^{(k)}) \| p_{\theta,2}(\cdot \mid b_1^{(k)})\Big).\end{aligned} \tag{10}$$

Sampling from q_1 yields an unbiased estimate of $\mathcal{L}_{12}$. Since the target is independent of θ, the gradient estimate is also unbiased (Appendix E.2).

The complete training objective is

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{NTP}} + \alpha\, \widetilde{\mathcal{L}}_{12}, \tag{11}$$

where $\alpha \geq 0$ controls alignment strength. The auxiliary loss supervises the first two decoding positions, while NTP remains active over the complete observed SID. The tokenizer and recommender architecture are unchanged.

3.4 PMI-Based Candidate Calibration

This section calibrates candidate scores using history-specific transition evidence relative to global prefix frequency. To reduce sensitivity to sparse counts, we shrink each source-conditioned distribution toward P_{global}. Let $a_k = \pi_{12}(v_{n-k+1})$; we define

$$\widetilde{T}(b \mid a_k) = \lambda_{a_k} T(b \mid a_k) + (1 - \lambda_{a_k}) P_{\mathrm{global}}(b), \qquad \widetilde{q}_{12}(b \mid \mathcal{H}) \quad = \sum_{k=1}^{H} w_k \widetilde{T}(b \mid a_k). \tag{12}$$

Here, $\lambda_{a_k} = C_{\mathrm{out}}(a_k) / \big(C_{\mathrm{out}}(a_k) + \tau\big)$ is a support-dependent reliability weight, and $\tau > 0$ controls the shrinkage strength. A frequently observed source prefix receives more weight from its transition distribution, whereas a prefix with limited support relies more on the global marginal. When $C_{\mathrm{out}}(a_k) = 0$, its adjusted distribution reduces to P_{global}. During training, such a source contributes the uniform transition row defined by Equation equation 5. For an empty history, we set $\widetilde{q}_{12}(\cdot \mid \mathcal{H}) = P_{\mathrm{global}}$, yielding zero calibration.

The calibration signal is

$$s_{\mathrm{PMI}}(b, \mathcal{H}) = \log \frac{\widetilde{q}_{12}(b \mid \mathcal{H})}{P_{\mathrm{global}}(b)}. \tag{13}$$

The signal is positive when the destination prefix is more likely under the current history than under its global reference frequency. It becomes zero when all source prefixes in the recent history have no observed outgoing transitions.

For each completed candidate $v \in \mathcal{B}(X)$, we use its sequence log-probability $m_v = \log p_\theta(c^v \mid X)$ as the base score for calibration

$$\mathrm{Score}(v) = m_v + \beta\, \mathrm{clip}\left(s_{\mathrm{PMI}}(\pi_{12}(v), \mathcal{H}), -\kappa, \kappa\right), \tag{14}$$

where $\beta \geq 0$ controls the calibration strength and $\kappa > 0$ bounds the magnitude of the PMI adjustment. Calibration only reranks the completed candidates and therefore leaves the candidate set and survival rates unchanged. Candidates sharing the same joint prefix receive the same adjustment.

Table 1: Recommendation performance on three benchmarks. H@K and N@K denote Hit@K and NDCG@K, respectively. Bold indicates the best result in each column.

		Beauty		Instruments		Yelp	
Category	Method	H@10	N@10	H@10	N@10	H@10	N@10
Sequential/ collaborative	MF (Koren et al., 2009)	.0474	.0191	.0735	.0412	.0381	.0190
	Caser (Tang & Wang, 2018)	.0347	.0176	.0710	.0409	.0263	.0134
	HGN (Ma et al., 2019)	.0512	.0266	.1048	.0774	.0326	.0159
	LightGCN (He et al., 2020)	.0511	.0260	.1000	.0728	.0407	.0207
	SASRec (Kang & McAuley, 2018)	.0588	.0313	.0947	.0690	.0296	.0152
	BERT4Rec (Sun et al., 2019)	.0347	.0170	.0822	.0608	.0291	.0159
Generative baselines	BIGRec (Bao et al., 2025)	.0299	.0198	.0576	.0491	.0169	.0142
	P5-SID (Geng et al., 2022; Fu et al., 2026)	.0584	.0335	.0964	.0730	.0324	.0170
	P5-CID (Geng et al., 2022; Fu et al., 2026)	.0597	.0347	.0987	.0751	.0347	.0181
	TIGER (Rajput et al., 2023)	.0610	.0331	.1058	.0797	.0407	.0213
Tokenizer/decoding improvements	STE (Huh et al., 2023)	.0134	.0067	.0554	.0360	.0147	.0077
	ETEGRec (Liu et al., 2025)	.0615	.0335	.1106	.0810	.0415	.0214
	LETTER (Wang et al., 2024)	.0671	.0347	.1107	.0825	.0429	.0228
	DIGER (Fu et al., 2026)	.0683	.0372	.1138	.0844	.0432	.0227
	APAO (Yu et al., 2026)	.0641	.0337	.1092	.0811	.0409	.0216
	BEAR (Yang et al., 2026)	.0817	.0444	.0988	.0737	**.0577**	**.0332**
Ours	PATH (no reranking)	.0844	.0477	.1167	.0874	.0515	.0286
Ours	PATH + PMI calibration	**.0856**	**.0484**	**.1176**	**.0881**	.0525	.0293

Computational cost. Let $P = |\mathcal{P}_{12}|$. An explicit dense transition table requires $O(P^2)$ storage, and directly constructing a history target costs $O(HP)$. Monte Carlo estimation reduces conditional model evaluations to K_{mc} sampled branches. Calibration requires $O(BH)$ table lookups and $O(B \log B)$ sorting, with no additional autoregressive decoding.

4 Experiments

We evaluate PATH from three perspectives: overall effectiveness, component contributions, and consistency with the proposed pruning mechanism. Main experiments compare recommendation performance across three benchmarks. Ablations isolate joint-prefix training and PMI calibration. Survival diagnostics, structure controls, and beam-width sweeps test whether the observed gains follow the early-pruning hypothesis, while sensitivity curves characterize the range of effective training and inference weights.

4.1 Experimental Setup

We evaluate on Amazon Beauty, Amazon Instruments, and Yelp using leave-one-out splitting and full-catalog Hit@10 and NDCG@10. PATH uses the T5-based LETTER+TIGER framework (Wang et al., 2024; Rajput et al., 2023), which also provides our NTP baseline. We rerun LETTER+TIGER, APAO (Yu et al., 2026), and BEAR (Yang et al., 2026) using their original settings, and the remaining baselines using the settings reported in DIGER (Fu et al., 2026). Our BEAR rerun fine-tunes Llama-3.2-3B. Unless otherwise stated, PATH and its NTP baseline use beam width $B = 20$. All experiments use seed 42. Calibration variants share the same checkpoint and candidate set within each dataset. PATH's alignment weight is selected on the validation set. Dataset statistics, detailed evaluation protocols, and implementation settings are provided in Appendix A.

4.2 Main Results

Table 1 reports recommendation performance across the three benchmarks. PATH with PMI calibration achieves the best Hit@10 and NDCG@10 on Beauty and Instruments, while BEAR performs best on Yelp. PATH reaches NDCG@10 values of 0.0484, 0.0881, and 0.0293, improving over LETTER+TIGER by 39.5%, 6.7%, and 28.6%, respectively. Compared with BEAR, PATH improves NDCG@10 by approximately 9.0% on Beauty and 19.5% on Instruments, with corresponding Hit@10 gains of 4.8% and 19.0%.

PATH outperforms BEAR on Beauty and Instruments but trails it on Yelp. This comparison involves different backbones and pretraining configurations: BEAR fine-tunes Llama-3.2-3B, while PATH

Table 2: Prefix and full-SID survival (%) at $B = 20$. Bold indicates the best result within each dataset.

Dataset	Training objective	α	S_{c_1}	S_{c_2}	S_{full}
Beauty	NTP	0	43.57	14.26	10.07
	c_1-only alignment	0.4	**46.39**	14.13	10.10
	Joint alignment	0.2	45.24	**15.76**	**11.75**
Instruments	NTP	0	47.02	16.83	13.84
	c_1-only alignment	0.2	**50.77**	16.79	13.80
	Joint alignment	0.2	49.27	**17.75**	**14.37**
Yelp	NTP	0	28.87	7.57	6.69
	c_1-only alignment	0.3	**32.60**	7.55	6.69
	Joint alignment	0.3	31.06	**8.58**	**7.67**

uses the T5-based LETTER+TIGER framework. BEAR's model capacity and pretraining may contribute to more effective modeling of Yelp interaction histories and partly explain the observed gap. PATH also achieves substantially faster inference than the BEAR configuration evaluated here, while its PMI calibration requires no additional autoregressive decoding. These comparisons reflect differences in backbone, pretraining, and training objectives.

Joint-prefix training accounts for most of PATH's improvement. Without calibration, PATH achieves NDCG@10 values of 0.0477, 0.0874, and 0.0286 on Beauty, Instruments, and Yelp, respectively. PMI calibration further improves ranking within the same candidate sets. Compared with DIGER, calibrated PATH improves NDCG@10 by 30.2%, 4.4%, and 28.9%, respectively. It also achieves higher Hit@10 and NDCG@10 than our APAO rerun on all three datasets. We further examine path retention and component contributions in sections 4.3 and 4.5, respectively.

4.3 Early Pruning and Joint-Prefix Survival

We first examine where targets leave the beam and whether improving the first decoding step translates into complete-path retention. For the NTP baseline with $B = 20$, first-token survival is 43.57%, 47.02%, and 28.87% on Beauty, Instruments, and Yelp, respectively. By the second step, these rates fall to 14.26%, 16.83%, and 7.57%. The fraction of Hit@10 misses whose targets are pruned within the first two decoding steps is

$$\rho_{\text{early}} = \frac{1 - S_{c_2}}{1 - \text{Hit@10}}, \tag{15}$$

which yields 91.9%, 93.5%, and 96.6% on the three datasets. These results identify the first two positions as the main location of target-path loss.

Table 2 compares NTP, c_1-only alignment, and joint alignment. The c_1-only variant improves first-token survival on all three datasets, while second-prefix and full-SID survival remain close to the baseline. On Beauty, for example, S_{c_1} increases from 43.57% to 46.39%, but S_{full} changes only from 10.07% to 10.10%.

Joint alignment achieves lower S_{c_1} than c_1-only alignment but improves both subsequent stages on every dataset. Its full-SID survival reaches 11.75%, 14.37%, and 7.67%, compared with 10.07%, 13.84%, and 6.69% for NTP. Thus, the configuration with the highest first-token survival does not produce the best complete-path retention. This supports supervising second-token continuations alongside first-token branches, as motivated in section 1. Complete survival and ranking diagnostics are provided in Table 7 in Appendix B.

4.4 History-Conditioned Supervision Controls

We assess the contribution of history matching on Beauty using the same joint-KL objective, training recipe, $\alpha = 0.2$, and $K_{\text{mc}} = 2$, with calibration disabled. The global-marginal control replaces the temporal target with P_{global}. The history-shuffled control cyclically shifts targets by one position within each batch, preserving the target collection while disrupting its correspondence with input histories.

Table 3: History-conditioned supervision controls on Beauty. Auxiliary losses use $\alpha = 0.2$, with inference calibration disabled. Survival rates are percentages. Bold indicates the best result in each column.

Training target	S_{c_1}	S_{c_2}	S_{full}	H@10	N@10
None (NTP)	43.57	14.26	10.07	.0671	.0347
Global marginal	40.95	13.56	9.44	.0667	.0368
History shuffled	40.88	13.37	9.39	.0669	.0360
History aligned (PATH)	**45.24**	**15.76**	**11.75**	**.0844**	**.0477**

Table 4: Training and calibration ablations. PATH calibration variants share the same checkpoint and candidate set within each dataset. Bold indicates the best result in each column.

		Beauty			Instruments			Yelp		
Training	Calibration	H@1	H@10	N@10	H@1	H@10	N@10	H@1	H@10	N@10
NTP	None	.0118	.0671	.0347	.0623	.1107	.0825	.0081	.0429	.0228
PATH	None	.0204	.0844	.0477	.0658	.1167	.0874	.0117	.0515	.0286
	Conditional prior	.0206	.0850	.0482	**.0664**	.1163	.0875	**.0123**	.0524	.0292
	PMI	**.0208**	**.0856**	**.0484**	**.0664**	**.1176**	**.0881**	.0118	**.0525**	**.0293**

As shown in Table 3, both controls improve NDCG@10 over NTP but reduce full-SID survival, whereas PATH improves both. At smaller alignment weights, the controls can also improve survival, but PATH achieves higher NDCG@10 across the reported settings (Table 8 in Appendix B).

4.5 Training and Inference Ablations

We examine the contributions of joint training and post-generation calibration. Table 4 compares NTP with PATH under three inference settings: no calibration, a conditional prior using $\log q_{12}$, and PMI calibration using $\log(\widetilde{q}_{12}/P_{\text{global}})$. Within each dataset, all PATH variants share the same checkpoint and completed candidate set.

Joint training provides most of the gain. Without calibration, PATH improves NDCG@10 over NTP by 37.5%, 5.9%, and 25.4% on Beauty, Instruments, and Yelp. The survival analysis in Section 4.3 also shows improved target-path retention.

Calibration provides an additional ranking gain. PMI increases both Hit@10 and NDCG@10 over uncalibrated PATH on every dataset. The complete PMI calibration scheme also yields slightly higher NDCG@10 than the conditional prior: .0484 versus .0482 on Beauty, .0881 versus .0875 on Instruments, and .0293 versus .0292 on Yelp. The advantage is metric-dependent: the conditional prior achieves higher Hit@1 on Yelp.

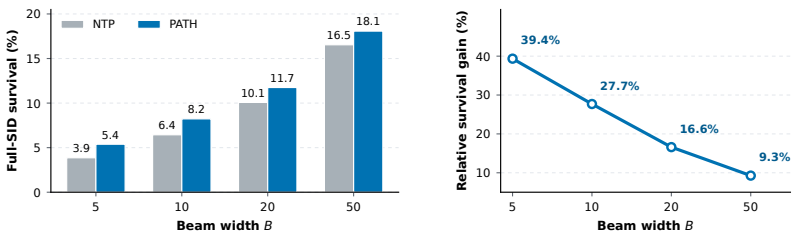
Calibration improves ranking within a fixed candidate set, leaving survival rates unchanged. The average rank of surviving targets also improves on all three datasets (Table 7 in Appendix B).

4.6 Beam-Width Analysis

We fix the NTP and PATH checkpoints on Beauty and vary $B \in \{5, 10, 20, 50\}$. PATH improves full-SID survival at every tested width (Figure 3). Relative gains range from 39.4% at $B = 5$ to 9.3% at $B = 50$, while absolute gains remain between 1.52 and 1.79 percentage points without a monotonic trend. Thus, narrower beams yield larger proportional improvements. Complete results appear in Appendix B, Table 9.

5 Related Work

Generative sequential recommendation. Sequential recommenders such as GRU4Rec (Hidasi et al., 2016), SASRec (Kang & McAuley, 2018), and BERT4Rec (Sun et al., 2019) learn to predict items from interaction histories. P5 (Geng et al., 2022) and M6-Rec (Cui et al., 2022) formulate recommendation tasks through language-based frameworks. TIGER (Rajput et al., 2023) combines



(a) Full-SID Survival **(b)** Relative Survival Gain

Figure 3: Beam-width analysis on Beauty using fixed NTP and PATH checkpoints. (a) Full-SID survival at $B \in \{5, 10, 20, 50\}$. (b) Relative survival gain of PATH over NTP. PATH improves survival at every tested width, with the largest relative gain at $B = 5$.

RQ-VAE-based semantic identifiers with autoregressive generation, providing the SID-based retrieval framework used in this work.

Semantic identifiers and recommendation alignment. LETTER (Wang et al., 2024) incorporates semantic, collaborative, and code-diversity regularization into item tokenization. LC-Rec (Zheng et al., 2024) combines vector-quantized item indices with alignment tasks to integrate language and collaborative semantics. ETEGRec (Liu et al., 2025) couples tokenization and recommendation through alignment objectives and alternating optimization, while DIGER (Fu et al., 2026) enables recommendation gradients to guide SID learning through differentiable tokenization. These methods improve item representations or their coordination with recommendation learning. PATH uses fixed SIDs and adds joint-prefix supervision and candidate calibration to the LETTER+TIGER framework.

Constrained decoding and prefix-aware optimization. Generative retrieval produces identifiers or retrieval keys autoregressively (Tay et al., 2022; Bevilacqua et al., 2022; Cao et al., 2021). Trie constraints restrict generation to valid identifiers, and STATIC (Su et al., 2026) accelerates this process through vectorized trie traversal. To address premature pruning, APAO (Yu et al., 2026) adaptively emphasizes vulnerable target prefixes, while BEAR (Yang et al., 2026) regularizes positive-token rankings using a relaxed necessary condition for beam retention. These methods focus on retaining the observed target path. PATH complements target-path supervision with a history-conditioned distribution over multiple joint prefixes, supervising both first-token branches and their second-token continuations.

Collaborative transition signals. Item associations and sequential transitions are established recommendation signals, as used in item-based collaborative filtering (Sarwar et al., 2001) and factorized personalized Markov chains (Rendle et al., 2010). Recent work also supplies collaborative cues to SID generation through personalized natural language at inference (Han et al., 2026). PATH aggregates training-corpus transitions at the joint-prefix level to construct history-conditioned soft targets. It reuses these statistics for post-generation calibration relative to global prefix frequency, connecting collaborative transition modeling with both autoregressive training and candidate ranking.

6 Conclusion and Limitations

We identify a gap between first-token survival and complete-path retention in generative recommendation. PATH addresses this gap by aligning early autoregressive predictions with history-conditioned joint-prefix targets derived from training transitions. An unbiased stochastic estimator makes this alignment tractable, while PMI calibration reuses the transition evidence to refine candidate ranking. Experiments across three benchmarks show improved path retention and recommendation accuracy, with most gains coming from joint-prefix training. Controls on Beauty further support the value of matching transition supervision to the current history. These findings motivate supervising early branches together with their continuations when learning to retrieve items through autoregressive decoding.

Limitations. First, the transition model uses first-order contextual transitions; higher-order and multihop temporal behavior remain open directions. Second, the experiments cover three medium-scale public datasets, and generalization to very large industrial catalogs requires further study.

AI USE STATEMENT

In this work, we utilized generative AI tools to assist in polishing the manuscript to enhance its clarity and readability. Additionally, we employed generative AI tools for literature search. Other relevant disclosure requirements are not applicable to this study.

We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide our source code and experimental configurations in an anonymous repository: `https://anonymous.4open.science/r/PATH-46A7`.

REFERENCES

Keqin Bao, Jizhi Zhang, Wenjie Wang, Yang Zhang, Zhengyi Yang, Yanchen Luo, Chong Chen, Fuli Feng, and Qi Tian. A bi-step grounding paradigm for large language models in recommendation systems. *ACM Transactions on Recommender Systems*, 3(4):1–27, 2025. doi: 10.1145/3716393.

Michele Bevilacqua, Giuseppe Ottaviano, Patrick Lewis, Wen tau Yih, Sebastian Riedel, and Fabio Petroni. Autoregressive search engines: Generating substrings as document identifiers. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 31668–31683, 2022. doi: 10.52202/068431-2296.

Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. Autoregressive entity retrieval. In *International Conference on Learning Representations (ICLR)*, 2021.

Zeyu Cui, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. M6-Rec: Generative pretrained language models are open-ended recommender systems. *arXiv preprint arXiv:2205.08084*, 2022. URL `https://arxiv.org/abs/2205.08084`.

Yizhou Dang, Enneng Yang, Yuting Liu, Guibing Guo, Linying Jiang, Xingwei Wang, and Jianzhe Zhao. Data augmentation for sequential recommendation: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 2026.

Junchen Fu, Xuri Ge, Alexandros Karatzoglou, Ioannis Arapakis, Suzan Verberne, Joemon M. Jose, and Zhaochun Ren. Differentiable semantic ID for generative recommendation. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 369–379, 2026. doi: 10.1145/3805712.3809641.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. Recommendation as language processing (RLP): A unified pretrain, personalized prompt & predict paradigm (P5). In *Proceedings of the 16th ACM Conference on Recommender Systems (RecSys)*, pp. 299–315, 2022.

Changjiang Han, Qingyang Li, Yaqiang Zang, Jikun Kang, Pinghua Gong, Xue Liu, and Bowei He. Restoring collaborative signals in semantic-ID generative recommendation via personalized natural language. *arXiv preprint arXiv:2607.27682*, 2026. URL `https://arxiv.org/abs/2607.27682`.

Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. LightGCN: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 639–648, 2020.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. In *International Conference on Learning Representations (ICLR)*, 2016.

Yupeng Hou, Jiacheng Li, Ashley Shin, Jinsung Jeon, Abhishek Santhanam, Wei Shao, Kaveh Hassani, Ning Yao, and Julian McAuley. Generating long semantic ids in parallel for recommendation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 956–966, 2025.

Minyoung Huh, Brian Cheung, Pulkit Agrawal, and Phillip Isola. Straightening out the straight-through estimator: Overcoming optimization challenges in vector quantized networks. In *International Conference on Machine Learning (ICML)*, pp. 14096–14113, 2023.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *IEEE International Conference on Data Mining (ICDM)*, pp. 197–206, 2018.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Enze Liu, Bowen Zheng, Cheng Ling, Lantao Hu, Han Li, and Wayne Xin Zhao. Generative recommender with end-to-end learnable item tokenization. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 729–739, 2025. doi: 10.1145/3726302.3729989.

Chen Ma, Peng Kang, and Xue Liu. Hierarchical gating networks for sequential recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 825–833, 2019.

Yuta Oshima, Masahiro Suzuki, Yutaka Matsuo, and Hiroki Furuta. Inference-time text-to-video alignment with diffusion latent beam search. *Advances in Neural Information Processing Systems*, 38:13170–13216, 2025.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, Maciej Kula, Ed Chi, and Maheswaran Sathiamoorthy. Recommender systems with generative retrieval. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, pp. 10299–10315, 2023. doi: 10.52202/075280-0452.

Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. Factorizing personalized Markov chains for next-basket recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*, pp. 811–820, 2010.

Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web (WWW)*, pp. 285–295, 2001.

Zhengyang Su, Isay Katsman, Yueqi Wang, Ruining He, Lukasz Heldt, Raghunandan Keshavan, Shao-Chuan Wang, Xinyang Yi, Mingyan Gao, Onkar Dalal, Lichan Hong, Ed H. Chi, and Ningren Han. Vectorizing the trie: Efficient constrained decoding for LLM-based generative retrieval on accelerators. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Volume 2 (KDD)*, pp. 8009–8020, 2026. doi: 10.1145/3770855.3818506.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 1441–1450, 2019.

Jiaxi Tang and Ke Wang. Personalized top-N sequential recommendation via convolutional sequence embedding. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM)*, pp. 565–573, 2018.

Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, Tal Schuster, William Cohen, and Donald Metzler. Transformer memory as a differentiable search index. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 21831–21843, 2022. doi: 10.52202/068431-1587.

Hao Wang, Wei Guo, Luankang Zhang, Jin Yao Chin, Yufei Ye, Huifeng Guo, Yong Liu, Defu Lian, Ruiming Tang, and Enhong Chen. Generative large recommendation models: Emerging trends in llms for recommendation. In *Companion Proceedings of the ACM on Web Conference 2025*, pp. 49–52, 2025.

Jingjin Wang and Jiawei Han. Proprag: Guiding retrieval with beam search over proposition paths. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 6212–6227, 2025.

Wenjie Wang, Honghui Bao, Xinyu Lin, Jizhi Zhang, Yongqi Li, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. Learnable item tokenization for generative recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 2400–2409, 2024. doi: 10.1145/3627673.3679569.

Weiqin Yang, Bohao Wang, Zhenxiang Xu, Jiawei Chen, Shengjia Zhang, Jingbang Chen, Canghong Jin, and Can Wang. BEAR: Towards beam-search-aware optimization for recommendation with large language models. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 2196–2207, 2026. doi: 10.1145/3805712.3809533.

Yuhao Yang, Chao Huang, Lianghao Xia, Chunzhen Huang, Da Luo, and Kangyi Lin. Debiased contrastive learning for sequential recommendation. In *Proceedings of the ACM web conference 2023*, pp. 1063–1073, 2023.

Yuanqing Yu, Yifan Wang, Weizhi Ma, Zhiqiang Guo, and Min Zhang. APAO: Bridging the training-inference gap in generative recommendation via adaptive prefix-aware optimization. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Volume 2 (KDD)*, pp. 6338–6349, 2026. doi: 10.1145/3770855.3818052.

Bowen Zheng, Yupeng Hou, Hongyu Lu, Yu Chen, Wayne Xin Zhao, Ming Chen, and Ji-Rong Wen. Adapting large language models by integrating collaborative semantics for recommendation. In *2024 IEEE 40th International Conference on Data Engineering (ICDE)*, pp. 1435–1448, 2024. doi: 10.1109/ICDE60146.2024.00118.

CONTENTS

1 Introduction **1**

2 Preliminaries **3**

2.1 Sequential Recommendation with Semantic Identifiers 3

2.2 Trie-Constrained Decoding and Prefix Survival 3

3 PATH: Joint-Prefix Alignment and PMI Calibration **3**

3.1 Joint-Prefix Transition Statistics 3

3.2 Recency-Weighted Temporal Targets 4

3.3 Joint Forward-KL Alignment 4

3.4 PMI-Based Candidate Calibration 5

4 Experiments **6**

4.1 Experimental Setup 6

4.2 Main Results 6

4.3 Early Pruning and Joint-Prefix Survival 7

4.4 History-Conditioned Supervision Controls 7

4.5 Training and Inference Ablations 8

4.6 Beam-Width Analysis 8

5 Related Work **8**

6 Conclusion and Limitations **9**

A Datasets and Implementation Details **14**

B Additional Survival and Control Results **14**

C Hyperparameter Sensitivity **15**

C.1 Alignment and Calibration Weights 15

C.2 Monte Carlo Sample Count 16

C.3 History Window and Temporal Decay 16

D Decoder-Only Evaluation **18**

E Mathematical Details **18**

E.1 Chain-Rule Decomposition of Joint KL 18

E.2 Unbiased Monte Carlo Estimation 19

A Datasets and Implementation Details

Table 5: Dataset statistics before validation and test holdout. Average length denotes the mean number of interactions per user.

Dataset	Users	Items	Interactions	Avg. length	Density
Beauty	22,363	12,101	198,502	8.88	0.0734%
Instruments	24,772	9,922	206,153	8.32	0.0839%
Yelp	30,431	20,033	316,354	10.40	0.0519%

Dataset statistics. Table 5 summarizes the datasets before validation and test holdout. Average sequence length is the number of interactions divided by the number of users. Density is the number of interactions divided by the product of the numbers of users and items.

Evaluation protocol. For each user, the penultimate interaction is reserved for validation and the final interaction for testing. Transition statistics are computed exclusively from training interactions. We report full-catalog Hit@10 and NDCG@10; with one held-out target per user, Hit@10 equals Recall@10. PATH and its NTP baseline use trie-constrained beam search with $B = 20$, unless otherwise stated. The beam-width analysis evaluates $B \in \{5, 10, 20, 50\}$.

Implementation and hyperparameters. LETTER+TIGER combines RQ-VAE semantic identifiers with a T5 recommender and uses codebook size $K = 256$. The tokenizer remains fixed during recommender training. The NTP baseline disables both alignment and calibration ($\alpha = 0, \beta = 0$).

Table 6 lists the main PATH settings. We select α separately for each dataset on the validation set from $\{0.1, 0.2, 0.3, 0.4, 0.5\}$. The history window and decay factor are shared by training-target construction and inference calibration. Models are trained independently across datasets, and all experiments conducted in this work use seed 42. Supplementary experiments specify any departures from the main configuration alongside their results.

Table 6: PATH hyperparameters for the main experiments. Dataset-specific alignment weights are listed in the order Beauty, Instruments, and Yelp.

Parameter	Main setting
History window K_{hist}	5
History decay γ	0.7
Transition smoothing ε	10^{-4}
Monte Carlo samples K_{mc}	2
Alignment weight α	0.2 / 0.2 / 0.3
Calibration weight β	0.02
Shrinkage strength τ	30
PMI clipping threshold κ	3.0

B Additional Survival and Control Results

Complete survival and ranking diagnostics. Table 7 extends the main survival comparison with recommendation metrics and the average target rank conditioned on full-SID survival. The joint-alignment and PMI rows use the same checkpoint within each dataset, so their survival rates coincide. Calibration affects the ranks of retained targets and the final recommendation metrics.

Control-weight sweeps. Table 8 reports the complete alignment-weight sweeps for the global-marginal and history-shuffled controls on Beauty. All auxiliary objectives use the same forward-KL formulation and $K_{\text{mc}} = 2$. Both controls provide gains at low weights but degrade at $\alpha = 0.4$. History-aligned PATH achieves higher NDCG@10 across its reported settings. The calibrated row connects these diagnostics to the main result; comparisons between target distributions use the uncalibrated rows.

Table 7: Complete survival and ranking diagnostics at $B = 20$. Survival rates are percentages. Rank denotes the average target rank conditioned on full-SID survival, for which lower is better. The PMI row calibrates the joint-alignment checkpoint immediately above it. Bold indicates the best value within each dataset.

Dataset	Configuration	S_{c_1}	S_{c_2}	S_{full}	Rank	Hit@10	NDCG@10
Beauty	NTP ($\alpha = 0$)	43.57	14.26	10.07	8.13	.0671	.0347
	c_1-only ($\alpha = 0.4$)	**46.39**	14.13	10.10	7.81	.0690	.0377
	Joint alignment ($\alpha = 0.2$)	45.24	**15.76**	**11.75**	7.21	.0844	.0477
	Joint alignment + PMI	45.24	**15.76**	**11.75**	**7.11**	**.0856**	**.0484**
Instruments	NTP ($\alpha = 0$)	47.02	16.83	13.84	5.31	.1107	.0825
	c_1-only ($\alpha = 0.2$)	**50.77**	16.79	13.80	5.40	.1101	.0818
	Joint alignment ($\alpha = 0.2$)	49.27	**17.75**	**14.37**	5.13	.1167	.0874
	Joint alignment + PMI	49.27	**17.75**	**14.37**	**5.02**	**.1176**	**.0881**
Yelp	NTP ($\alpha = 0$)	28.87	7.57	6.69	8.29	.0429	.0228
	c_1-only ($\alpha = 0.3$)	**32.60**	7.55	6.69	8.23	.0435	.0234
	Joint alignment ($\alpha = 0.3$)	31.06	**8.58**	**7.67**	7.82	.0515	.0286
	Joint alignment + PMI	31.06	**8.58**	**7.67**	**7.61**	**.0525**	**.0293**

Table 8: Control-weight sweeps on Beauty with $B = 20$ and seed 42. ΔN@10 denotes the relative NDCG@10 change from the NTP baseline. All rows except the final row disable inference calibration. Survival rates are percentages. Bold indicates the best result in each metric column.

Target / configuration	α	S_{c_1}	S_{c_2}	S_{full}	Hit@10	NDCG@10	ΔN@10
NTP	0	43.57	14.26	10.07	.0671	.0347	0.0%
Global marginal	0.1	42.80	14.30	10.22	.0715	.0382	+10.2%
	0.2	40.95	13.56	9.44	.0667	.0368	+6.1%
	0.4	37.28	11.75	7.63	.0567	.0296	-14.7%
History shuffled	0.1	42.62	14.40	10.20	.0724	.0386	+11.3%
	0.2	40.88	13.37	9.39	.0669	.0360	+3.8%
	0.4	35.87	11.06	7.13	.0512	.0269	-22.5%
History aligned (PATH)	0.2	45.24	**15.76**	**11.75**	.0844	.0477	+37.5%
	0.4	**45.58**	15.33	11.54	.0806	.0455	+31.2%
PATH + PMI	0.2	45.24	**15.76**	**11.75**	**.0856**	**.0484**	**+39.5%**

Complete beam-width results. Table 9 provides the numerical results underlying Figure 3. Both models use fixed checkpoints on Beauty, with $\alpha = 0.2$ for PATH. Wider beams increase survival for both models, while PATH retains more targets at every tested width.

Table 9: Prefix survival at different beam widths on Beauty. Survival rates are percentages. Relative gain refers to PATH's full-SID survival relative to NTP at the same beam width. Bold indicates higher survival within each pair.

Beam B	Model	S_{c_1}	S_{c_2}	S_{full}	Relative gain
5	NTP	27.38	7.12	3.86	–
	PATH	**28.95**	**8.49**	**5.38**	+39.4%
10	NTP	35.02	10.19	6.44	–
	PATH	**36.48**	**11.76**	**8.23**	+27.7%
20	NTP	43.57	14.26	10.07	–
	PATH	**45.24**	**15.76**	**11.75**	+16.6%
50	NTP	55.84	21.40	16.54	–
	PATH	**58.69**	**22.84**	**18.07**	+9.3%

C Hyperparameter Sensitivity

C.1 Alignment and Calibration Weights

Figure 4 reports test NDCG@10 across the training alignment weight α and inference calibration weight β. The α sweep evaluates uncalibrated models over the tested grid in $[0, 0.5]$. The highest

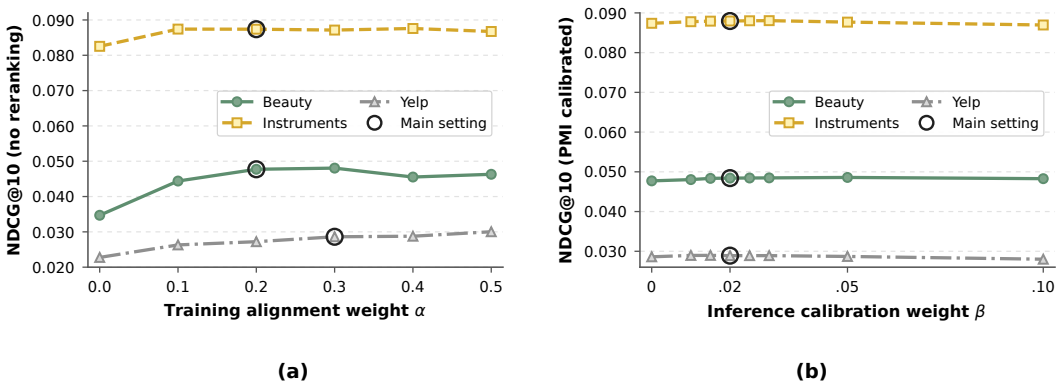


Figure 4: Sensitivity to alignment and calibration weights on three datasets. (a) Uncalibrated NDCG@10 versus α. (b) PMI-calibrated NDCG@10 versus β. Outlined markers indicate the main settings, $\alpha = 0.2/0.2/0.3$ and $\beta = 0.02$, rather than the maxima of the test curves.

observed values occur at $\alpha = 0.3$, 0.4, and 0.5 for Beauty, Instruments, and Yelp, respectively. Yelp is still improving at the upper boundary, so the sweep does not establish an interior optimum.

The main alignment weights are selected on the validation set and therefore need not coincide with the maxima of these test curves. Outlined markers indicate the main settings $\alpha = 0.2/0.2/0.3$.

For inference calibration, we evaluate the tested β grid over $[0, 0.10]$. The observed maxima occur at 0.05, 0.03, and 0.015 for Beauty, Instruments, and Yelp, respectively. The common setting $\beta = 0.02$ is within 0.00017 NDCG@10 of the highest reported value on each dataset.

C.2 Monte Carlo Sample Count

The conditional term in equation equation 9 is estimated using samples from the first-token target distribution. We examine $K_{\text{mc}} \in \{2, 3, 4, 5, 6\}$ on Instruments while fixing $\alpha = 0.2$, $K_{\text{hist}} = 5$, $\gamma = 0.7$, and seed 42. The $K_{\text{mc}} = 2$ row reuses the main-experiment checkpoint; the remaining settings are trained independently. All settings are evaluated with $\beta = 0.02$ and $\tau = 30$.

As shown in Table 10, increasing the sample count does not produce monotonic gains. NDCG@10 ranges from .0871 to .0891. The smallest tested count, $K_{\text{mc}} = 2$, achieves .0881, compared with .0891 at $K_{\text{mc}} = 5$. This sweep supports using a small number of sampled conditional evaluations, although larger sample counts can improve the observed result.

Table 10: Monte Carlo sample-count sensitivity on Instruments. All configurations use $\alpha = 0.2$, $K_{\text{hist}} = 5$, $\gamma = 0.7$, and seed 42, with PMI parameters $\beta = 0.02$ and $\tau = 30$. Full-SID survival is reported as a percentage. Bold indicates the best result in each column.

K_{mc}	H@10	N@10	S_{full}
2	.1176	.0881	14.37
3	.1171	.0880	14.34
4	.1163	.0878	14.46
5	**.1194**	**.0891**	**14.66**
6	.1156	.0871	14.31

C.3 History Window and Temporal Decay

We vary the history window K_{hist} and decay factor γ on Beauty, retraining each configuration from scratch. The window sweep fixes $\gamma = 0.7$, and the decay sweep fixes $K_{\text{hist}} = 5$. All runs use $\alpha = 0.3$, $K_{\text{mc}} = 4$, $\beta = 0.02$, $\tau = 30$, and seed 42.

These settings change the history aggregation used in both training and inference calibration, so Hit@10 and NDCG@10 measure end-to-end sensitivity. Full-SID survival is measured before calibration and reflects the trained model's candidate generation.

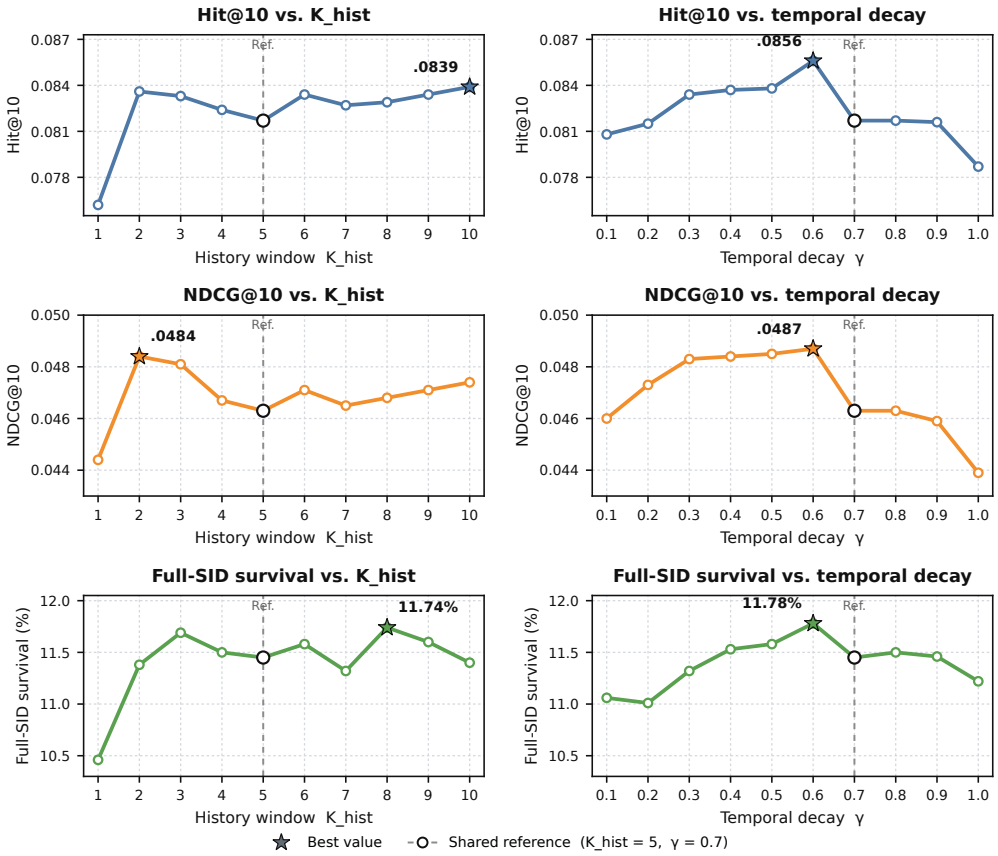


Figure 5: History-window and temporal-decay sensitivity on Beauty. Every configuration is retrained with $\alpha = 0.3$, $K_{\text{mc}} = 4$, and seed 42, then evaluated with $\beta = 0.02$ and $\tau = 30$. The left column varies K_{hist} with $\gamma = 0.7$; the right column varies γ with $K_{\text{hist}} = 5$. The rows report Hit@10, NDCG@10, and full-SID survival, respectively. Stars indicate the best value in each sweep. Gray dashed lines and open circles mark the shared reference configuration $(K_{\text{hist}}, \gamma) = (5, 0.7)$. Full-SID survival is reported as a percentage.

Figure 5 shows that using only the most recent interaction gives the lowest NDCG@10 in the window sweep. Aggregating at least two interactions improves the reported metrics, but larger windows do not provide monotonic gains. The highest NDCG@10 occurs at $K_{\text{hist}} = 2$, whereas the highest full-SID survival occurs at $K_{\text{hist}} = 8$.

For temporal decay, $\gamma \in [0.3, 0.6]$ yields the highest NDCG@10 values in this configuration. Uniform weighting ($\gamma = 1$) gives .0439, compared with .0487 at $\gamma = 0.6$. These observations support combining several recent interactions while discounting older ones in the tested setting.

D DECODER-ONLY EVALUATION

Table 11: Decoder-only evaluation with $B = 20$ and seed 42. Joint alignment uses $\alpha = 0.4$ and $K_{\mathrm{mc}} = 2$ across all datasets. PMI calibration is disabled. Full-SID survival is reported as a percentage. Bold indicates the better result within each dataset.

Dataset	Training objective	H@10	N@10	S_{full}
Beauty	NTP	.0557	.0302	8.36
	Joint alignment	**.0605**	**.0341**	**8.71**
Instruments	NTP	.1004	**.0759**	12.32
	Joint alignment	**.1052**	.0754	**13.02**
Yelp	NTP	.0317	.0165	5.02
	Joint alignment	**.0351**	**.0188**	**5.48**

We evaluate whether joint-prefix alignment also improves target retention in a decoder-only recommender. The model is an eight-layer LLaMA-style architecture with hidden size 128, eight attention heads, and intermediate size 1024. It is trained from scratch with effective batch size 1024, learning rate 5×10^{-4}, bfloat16 precision, and seed 42.

Within each dataset, the NTP and joint-alignment runs share the data split, SID tokenizer, and trie-constrained beam search with $B = 20$. Joint alignment uses $\alpha = 0.4$ and $K_{\mathrm{mc}} = 2$ on all three datasets. PMI calibration is disabled.

Table 11 shows that joint alignment improves Hit@10 and full-SID survival on all three datasets. NDCG@10 increases from .0302 to .0341 on Beauty and from .0165 to .0188 on Yelp, but decreases slightly from .0759 to .0754 on Instruments. These results support the transfer of the path-retention benefit to the tested decoder-only architecture, while showing that improved survival does not guarantee higher NDCG@10.

E MATHEMATICAL DETAILS

We derive the joint-KL decomposition and establish the unbiasedness and sampling variance of the auxiliary-loss estimator. The analysis fixes a training example with interaction history $\mathcal{H}$ and SID-encoded input X. The tokenizer, transition table, and target-construction hyperparameters are fixed, so the complete target distribution $q_{12}(\cdot \mid \mathcal{H})$ is independent of the recommender parameters θ.

E.1 CHAIN-RULE DECOMPOSITION OF JOINT KL

Notation and support. For readability, we suppress $\mathcal{H}$ and X and write

$$\begin{aligned} q_{12}(b_1, b_2) &= q_{12}(b_1, b_2 \mid \mathcal{H}), \\ q_1(b_1) &= q_1(b_1 \mid \mathcal{H}), \\ q_2(b_2 \mid b_1) &= q_2(b_2 \mid b_1, \mathcal{H}), \\ p_{\theta,1}(b_1) &= p_{\theta,1}(b_1 \mid X), \\ p_{\theta,2}(b_2 \mid b_1) &= p_{\theta,2}(b_2 \mid X, b_1). \end{aligned}$$

The target q_{12} is a normalized distribution on the finite vocabulary $C_1 \times C_2$, extended by zero outside $\mathcal{P}_{12}$.

Define its first-token marginal and active first-token support as

$$q_1(b_1) = \sum_{b_2 \in C_2} q_{12}(b_1, b_2), \qquad \mathcal{A} = \{b_1 \in C_1 : q_1(b_1) > 0\}. \tag{16}$$

For $b_1 \in \mathcal{A}$, the conditional target is

$$q_2(b_2 \mid b_1) = \frac{q_{12}(b_1, b_2)}{q_1(b_1)}, \qquad q_{12}(b_1, b_2) = q_1(b_1) q_2(b_2 \mid b_1). \tag{17}$$

Branches outside $\mathcal{A}$ have zero joint target mass and are excluded from the conditional expectations below. No conditional target needs to be evaluated on those branches.

The model distributions are normalized over their corresponding token vocabularies and satisfy

$$p_{\theta,12}(b_1, b_2) = p_{\theta,1}(b_1)\, p_{\theta,2}(b_2 \mid b_1). \tag{18}$$

If token masks are used, these symbols refer to the resulting normalized distributions used in the training objective.

We assume $p_{\theta,12}(b_1, b_2) > 0$ whenever $q_{12}(b_1, b_2) > 0$. On the finite vocabularies, this ensures that the marginal KL and every conditional KL on $\mathcal{A}$ are finite. Terms with zero target probability contribute zero and are omitted from logarithmic sums.

Proposition 1 (Joint-KL decomposition). Under these definitions,

$$\begin{aligned}\mathrm{KL}(q_{12} \| p_{\theta,12}) = \mathrm{KL}(q_1 \| p_{\theta,1}) \\ + \sum_{b_1 \in \mathcal{A}} q_1(b_1) \mathrm{KL}(q_2(\cdot \mid b_1) \| p_{\theta,2}(\cdot \mid b_1))\,.\end{aligned} \tag{19}$$

Proof. Substituting the target and model factorizations into the definition of KL gives

$$\begin{aligned}\mathrm{KL}(q_{12} \| p_{\theta,12}) &= \sum_{b_1 \in \mathcal{A}} \sum_{b_2 \in C_2} q_1(b_1) q_2(b_2 \mid b_1) \log \frac{q_1(b_1) q_2(b_2 \mid b_1)}{p_{\theta,1}(b_1) p_{\theta,2}(b_2 \mid b_1)} \\ &= \sum_{b_1 \in \mathcal{A}} \sum_{b_2 \in C_2} q_1(b_1) q_2(b_2 \mid b_1) \log \frac{q_1(b_1)}{p_{\theta,1}(b_1)} \\ &\quad + \sum_{b_1 \in \mathcal{A}} \sum_{b_2 \in C_2} q_1(b_1) q_2(b_2 \mid b_1) \log \frac{q_2(b_2 \mid b_1)}{p_{\theta,2}(b_2 \mid b_1)}.\end{aligned} \tag{20}$$

For each $b_1 \in \mathcal{A}$, $\sum_{b_2 \in C_2} q_2(b_2 \mid b_1) = 1$. The first term therefore reduces to

$$\sum_{b_1 \in \mathcal{A}} q_1(b_1) \log \frac{q_1(b_1)}{p_{\theta,1}(b_1)} = \mathrm{KL}(q_1 \| p_{\theta,1}). \tag{21}$$

The second term is

$$\sum_{b_1 \in \mathcal{A}} q_1(b_1) \mathrm{KL}(q_2(\cdot \mid b_1) \| p_{\theta,2}(\cdot \mid b_1))\,. \tag{22}$$

Combining the two terms establishes equation equation 19.

Implication for joint alignment. The conditional term is nonnegative, so

$$\mathrm{KL}(q_{12} \| p_{\theta,12}) \geq \mathrm{KL}(q_1 \| p_{\theta,1}). \tag{23}$$

Even when $q_1 = p_{\theta,1}$, mismatched second-token continuations can leave a positive joint KL. Matching the first-token marginal alone therefore does not ensure that the joint distributions match. The conditional component supplies the additional supervision needed to align continuations within each target-supported branch. Its effect on finite-beam target survival is evaluated experimentally in section 4.3.

E.2 Unbiased Monte Carlo Estimation

Estimator. Define the exact marginal term and the conditional loss associated with an active first-token branch as

$$\mathcal{L}_1(\theta) = \mathrm{KL}(q_1 \| p_{\theta,1}), \tag{24}$$

$$f_\theta(b_1) = \mathrm{KL}(q_2(\cdot \mid b_1) \| p_{\theta,2}(\cdot \mid b_1))\,, \qquad b_1 \in \mathcal{A}. \tag{25}$$

The chain-rule decomposition gives

$$\mathcal{L}_{12}(\theta) = \mathcal{L}_1(\theta) + \sum_{b_1 \in \mathcal{A}} q_1(b_1) f_\theta(b_1). \tag{26}$$

For an integer $K_{\mathrm{mc}} \geq 1$, draw independent samples with replacement:

$$b_1^{(1)}, \ldots, b_1^{(K_{\mathrm{mc}})} \overset{\text{i.i.d.}}{\sim} q_1. \tag{27}$$

All sampled branches belong to $\mathcal{A}$ with probability one. The estimator is

$$\widetilde{\mathcal{L}}_{12}(\theta) = \mathcal{L}_1(\theta) + \frac{1}{K_{\mathrm{mc}}} \sum_{k=1}^{K_{\mathrm{mc}}} f_\theta(b_1^{(k)}). \tag{28}$$

The marginal term is computed exactly. For each sampled branch, the conditional KL is also computed exactly over the second-token vocabulary. Sampling approximates only the expectation over first-token branches.

All expectations and variances below are taken over these branch samples, with the training example and model distributions held fixed.

Proposition 2 (Unbiased objective and gradient). The estimator satisfies

$$\mathbb{E}\left[\widetilde{\mathcal{L}}_{12}(\theta)\right] = \mathcal{L}_{12}(\theta). \tag{29}$$

If the model probabilities on the target support are differentiable and positive in a neighborhood of the parameter value under consideration, then

$$\mathbb{E}\left[\nabla_\theta \widetilde{\mathcal{L}}_{12}(\theta)\right] = \nabla_\theta \mathcal{L}_{12}(\theta). \tag{30}$$

Proof of objective unbiasedness. Each sampled branch has distribution q_1. Consequently,

$$\mathbb{E}\left[f_\theta(b_1^{(k)})\right] = \sum_{b_1 \in \mathcal{A}} q_1(b_1) f_\theta(b_1). \tag{31}$$

By linearity of expectation,

$$\begin{aligned} \mathbb{E}\left[\widetilde{\mathcal{L}}_{12}(\theta)\right] &= \mathcal{L}_1(\theta) + \frac{1}{K_{\mathrm{mc}}} \sum_{k=1}^{K_{\mathrm{mc}}} \mathbb{E}\left[f_\theta(b_1^{(k)})\right] \\ &= \mathcal{L}_1(\theta) + \sum_{b_1 \in \mathcal{A}} q_1(b_1) f_\theta(b_1) \\ &= \mathcal{L}_{12}(\theta). \end{aligned} \tag{32}$$

Proof of gradient unbiasedness. The entire target distribution is independent of θ, so q_1, q_2, and $\mathcal{A}$ remain fixed during differentiation. Because $\mathcal{A}$ is finite and each branch loss is differentiable,

$$\nabla_\theta \mathcal{L}_{12}(\theta) = \nabla_\theta \mathcal{L}_1(\theta) + \sum_{b_1 \in \mathcal{A}} q_1(b_1) \nabla_\theta f_\theta(b_1). \tag{33}$$

Treating sampled token indices as constants during differentiation gives

$$\nabla_\theta \widetilde{\mathcal{L}}_{12}(\theta) = \nabla_\theta \mathcal{L}_1(\theta) + \frac{1}{K_{\mathrm{mc}}} \sum_{k=1}^{K_{\mathrm{mc}}} \nabla_\theta f_\theta(b_1^{(k)}). \tag{34}$$

Taking expectations,

$$\begin{aligned} \mathbb{E}\left[\nabla_\theta \widetilde{\mathcal{L}}_{12}(\theta)\right] &= \nabla_\theta \mathcal{L}_1(\theta) + \frac{1}{K_{\mathrm{mc}}} \sum_{k=1}^{K_{\mathrm{mc}}} \mathbb{E}\left[\nabla_\theta f_\theta(b_1^{(k)})\right] \\ &= \nabla_\theta \mathcal{L}_1(\theta) + \sum_{b_1 \in \mathcal{A}} q_1(b_1) \nabla_\theta f_\theta(b_1) \\ &= \nabla_\theta \mathcal{L}_{12}(\theta). \end{aligned} \tag{35}$$

No derivative of the sampling distribution is required because it is independent of θ.

The result holds for every $K_{\mathrm{mc}} \geq 1$. Choosing $K_{\mathrm{mc}} = 2$ changes computational cost and sampling variance, but not the expected auxiliary objective or gradient.

Combined training objective. For a fixed alignment weight α, define the exact and sampled training objectives as

$$\mathcal{J}(\theta) = \mathcal{L}_{\mathrm{NTP}}(\theta) + \alpha \mathcal{L}_{12}(\theta), \tag{36}$$

$$\widetilde{\mathcal{J}}(\theta) = \mathcal{L}_{\mathrm{NTP}}(\theta) + \alpha \widetilde{\mathcal{L}}_{12}(\theta). \tag{37}$$

The NTP term is unaffected by branch sampling. Therefore,

$$\mathbb{E}\left[\widetilde{\mathcal{J}}(\theta)\right] = \mathcal{J}(\theta), \qquad \mathbb{E}\left[\nabla_\theta \widetilde{\mathcal{J}}(\theta)\right] = \nabla_\theta \mathcal{J}(\theta), \tag{38}$$

where the gradient identity assumes that $\mathcal{L}_{\mathrm{NTP}}$ is differentiable.

Sampling variance. For fixed θ and a fixed training example, let

$$\mu_f = \mathbb{E}_{b_1 \sim q_1}[f_\theta(b_1)], \qquad \sigma_f^2 = \mathbb{E}_{b_1 \sim q_1}\left[(f_\theta(b_1) - \mu_f)^2\right]. \tag{39}$$

These moments are finite because $\mathcal{A}$ is finite and each branch loss is finite.

The marginal term contributes no branch-sampling variance. Independence of the sampled branches eliminates cross-sample covariance terms, giving

$$\begin{aligned} \mathrm{Var}\left[\widetilde{\mathcal{L}}_{12}\right] &= \mathrm{Var}\left[\frac{1}{K_{\mathrm{mc}}} \sum_{k=1}^{K_{\mathrm{mc}}} f_\theta(b_1^{(k)})\right] \\ &= \frac{1}{K_{\mathrm{mc}}^2} \sum_{k=1}^{K_{\mathrm{mc}}} \mathrm{Var}\left[f_\theta(b_1^{(k)})\right] \\ &= \frac{\sigma_f^2}{K_{\mathrm{mc}}}. \end{aligned} \tag{40}$$

The same reasoning applies to the gradient. Let

$$g_\theta(b_1) = \nabla_\theta f_\theta(b_1), \qquad \Sigma_g = \mathrm{Cov}_{b_1 \sim q_1}[g_\theta(b_1)]. \tag{41}$$

The covariance exists under the differentiability and positivity assumptions above because the active support is finite. Since the marginal gradient is deterministic with respect to branch sampling,

$$\begin{aligned} \mathrm{Cov}\left[\nabla_\theta \widetilde{\mathcal{L}}_{12}\right] &= \mathrm{Cov}\left[\frac{1}{K_{\mathrm{mc}}} \sum_{k=1}^{K_{\mathrm{mc}}} g_\theta(b_1^{(k)})\right] \\ &= \frac{\Sigma_g}{K_{\mathrm{mc}}}. \end{aligned} \tag{42}$$

For the combined objective,

$$\mathrm{Cov}\left[\nabla_\theta \widetilde{\mathcal{J}}\right] = \frac{\alpha^2 \Sigma_g}{K_{\mathrm{mc}}}. \tag{43}$$

These identities quantify branch-sampling noise at fixed model parameters. They do not include other sources of training randomness, such as minibatch selection or dropout. Increasing K_{mc} reduces this component of estimator variance, but does not imply monotonic improvements in final recommendation metrics, which also depend on the optimization trajectory.