Token-Level Debiasing in LLM for Personalized Recommendation: An Information Gain Approach

Zijie Lin
NUS
zijie.lin@u.nus.edu

Abstract

Large Language Models (LLMs) struggle with recommendation tasks due to token-level biases: they overfit to low-information tokens while neglecting meaningful ones. We frame recommendation as a decision process, measuring token importance via Information Gain (IG), and identify two key failure modes—zero-IG token dominance in training and low-IG bias in decoding. Our IG-based Debiasing (IGD) method reweights tokens dynamically during fine-tuning and beam search, improving accuracy and diversity without auxiliary models. Experiments show consistent gains over state-of-the-art approaches.

1 Introduction

Recommendation systems are essential to help users discover relevant and personalized content. Recently, Large Language Models (LLMs) have demonstrated exceptional text comprehension and reasoning capabilities, emerging as powerful tools for recommendation tasks (Harte et al., 2023; Lin et al., 2025). A leading approach involves using LLMs as the core recommender engine, taking users' historical interactions and current task requirements as prompt inputs, to predict the next appropriate item (Liao et al., 2024; Liu et al., 2024). This methodology has shown remarkable effectiveness in understanding user preferences and generating customized recommendations (Wu et al., 2024; Zhao et al., 2024).

Since the objective of LLM pre-training is not designed for recommendation tasks, researchers have explored various adaptation techniques using empirical recommendation data to enhance LLMs' recommendation performance (Lin et al., 2024b; Yue et al., 2023). A common approach is supervised fine-tuning (SFT), where user interaction history and task specifications serve as input instructions, and the next selected item functions as the label (Zhang et al., 2023; Bao et al., 2023a; Lin et al., 2024a). After adaptation, LLMs typically generate recommendations using beam search decoding, followed by logic-based re-ranking to produce the top-K items (Bao et al., 2024; Zhang et al., 2024; Gao et al., 2024; 2025).

While the SFT-Beam Search paradigm leverages LLMs to downstream item recommendation task in a natural way, there exists a fundamental mismatch between text generation tasks and recommendation objectives (Bao et al., 2023b). Items, usually represented by their titles, are sequences of tokens. Although LLMs should optimize the overall probability of a desired item, SFT operates at the token level, leading to an imbalance in the token-level probability distribution. This creates two token-level issues:

- **Importance–Probability Contradiction:** LLMs tend to assign excessively high probabilities (close to 1) to tokens with minimal semantic value, such as prepositions or punctuation (Bao et al., 2024). Conversely, tokens that carry meaningful semantic information often receive relatively low probabilities. This creates a contradiction between token importance and generation probability. As a consequence, LLMs prioritize recommending items with more low-semantic value tokens and less meaningful tokens.
- **Homogeneity Bias:** LLM-generated recommendations frequently begin with common tokens (e.g., movie recommendations often start with "The"). These common

tokens shared across many items receive disproportionate attention during fine-tuning, while rare but distinctive tokens remain under-learned (Bao et al., 2023b; Gao et al., 2024). This bias reduces recommendation diversity and limits the model's ability to capture unique item characteristics.

Existing research has attempted to alleviate the importance–probability contradiction issue through token loss reweighting techniques during supervised fine-tuning (SFT). Zhang et al. (2024) proposes positional normalization methods that assign higher weights to tokens in the prefix and reduce weights for tokens near the end. Additionally, it introduces causal fine-tuning, which emphasizes distinctive tokens that are more sensitive to input content.

To improve recommendation diversity and mitigate homogeneity bias, a common approach involves using a traditional model to reward tokens that align with its predictions during fine-tuning (Gao et al., 2025) or decoding (Bao et al., 2024). Although effective, these methods constrain output to match the auxiliary model and may not scale effectively with increasingly powerful LLMs. A novel alternative employs self-play and Direct Preference Optimization (DPO) after each SFT epoch (Gao et al., 2024). Although this approach successfully improves diversity, it suffers from probability collapse.

All these approaches aim to emphasize tokens meaningful for recommendation tasks while reducing the influence of less relevant ones. While their methods are based on the intuitive belief that **tokens with higher uncertainty carry more information** (Bao et al., 2024; Zhang et al., 2024), they do not thoroughly investigate the underlying measures of uncertainty. As a result, when their assumptions fail, they risk introducing or even amplifying biases.

To address these limitations, we propose modeling LLM generation as a decision process. A prefix token sequence features uncertainty quantified by entropy. Each token's contribution to the final result can be measured by the reduction of this uncertainty, i.e., Information Gain (IG). Our analysis reveals that:

- 1. For recommendation tasks, tokens with zero-IG (i.e., those that do not contribute to deciding the recommended items) dominate the output. These tokens quickly reach nearly zero loss during SFT and demonstrate extremely high token likelihood during decoding, confirming previous findings (Bao et al., 2024).
- 2. During decoding, LLMs tend to favor low IG (common, non-distinctive) tokens, leading to homogeneity bias.

Based on these insights, we develop a novel plug-and-play approach that leverages information gain during both fine-tuning and decoding phases to evaluate token decisiveness and recalibrate token weights accordingly. This approach enables more precise control over the generation process while maintaining the inherent strengths of LLMs in recommendation tasks.

Our contributions are as follows:

- 1. We analyze token-level logical imbalances in current LLM recommendation paradigms and identify limitations in existing token reweighting methods.
- 2. Through decision modeling, we uncover token-level logical inconsistencies in SFT and decoding biases that significantly hinder LLM performance in recommendation tasks.
- 3. We introduce an information gain-based debiasing (IGD) reweighting strategy that addresses these issues during both the SFT and decoding stages.
- 4. Experimental results demonstrate the superior performance of our proposed method compared to existing approaches.

2 Related Work

Previous studies have made important strides in addressing the challenges of using LLMs for recommendation systems, though significant limitations remain. Here, we examine key approaches and their constraints.

D3 identified the phenomenon of "ghost tokens" - tokens that contribute little to distinguishing between items but receive disproportionate attention during generation. Their finding that setting length penalty to zero improves performance suggests that focusing on truly decisive tokens is crucial for bridging the language and item spaces. However, their approach of using LLM uncertainty as a proxy for identifying ghost tokens has a fundamental flaw: **high model certainty about a token could either indicate it's a non-informative ghost token OR that the model has learned a strong, correct association.** This ambiguity limits the effectiveness of LLM uncertainty-based approaches.

Positional decaying techniques have been adopted in LLM fine-tuning for recommendation tasks, based on **the assumption that earlier tokens carry more weight in determining the final item. While this assumption often holds, it fails to account for common prefix tokens (like articles "The" or "A") that contribute minimal semantic value despite their position.** Consequently, positional decaying could exaggerate the homogeneity bias. This highlights the need for more nuanced approaches to token importance that consider semantic contribution rather than just position.

Recent work on integrated grounding reweighting attempts to address distribution mismatches by adjusting item scores post-generation to better align with real item distributions. **While this approach shows promise in improving final recommendations, its post-hoc nature means it can only partially mitigate biases that have already been introduced during the decoding phase.** The token-level biases that accumulate during generation continue to limit the diversity of the initial candidate set.

These limitations point to two critical research gaps that our work addresses:

1. The need for a principled method to **measure token decisiveness** that doesn't rely on problematic proxies like model uncertainty or position

2. The importance of **debiasing throughout the entire recommendation pipeline**, not just as a post-processing step

3 Preliminary

This section introduces the framework for analyzing LLM-based recommendation systems. We first formalize the supervised fine-tuning (SFT) process that bridges item and language spaces, then examine the beam search decoding mechanism. Through this analysis, we identify potential bias introduced by current approaches.

3.1 Supervised Fine-tuning

Converting recommendation tasks into language generation requires mapping items to their textual representations. Given an input prompt x (e.g., user preferences or query) and target sequence y (item descriptions), the LLM is fine-tuned using the following objective:

$$\mathcal{L} = \sum_{t=1}^{|y|} l(f_\theta(x, y_{<t}); y_t), \tag{1}$$

where f_θ denotes the LLM with parameters θ, $y_{<t}$ represents the token sequence preceding position t, and l is the cross-entropy loss. This formulation treats recommendation as a token-by-token generation task, optimizing the model to predict each subsequent token given the context.

3.2 Generation Mechanism

During inference, recommendations are generated through autoregressive token prediction. The probability of generating a complete item description y given input x follows the chain rule of probability:

$$p(y|x) = \prod_{i=1}^{m} p(y_i|x, y_{<i}), \tag{2}$$

where m denotes the sequence length. This decomposition enables step-wise generation but introduces dependencies on token-level probabilities.

3.3 Beam Search Decoding

To balance exploration and computational efficiency, beam search maintains k most promising partial sequences at each generation step. The instantaneous score for a candidate token y_t is defined as:

$$S(y_t) = \log(p(y_t|x, y_{\leq t-1})) \tag{3}$$

The cumulative sequence score is computed recursively:

$$S(y_{\leq t}) = S(y_{\leq t-1}) + S(y_t) \tag{4}$$

This scoring mechanism prioritizes sequences with consistently high token probabilities, rather than those containing particularly informative or decisive tokens.

3.4 Structural Limitations

The interaction between token-level optimization and beam search introduces three systemic issues:

1. Objective Misalignment: While fine-tuning optimizes token-level prediction accuracy, and recommendation quality depends on generating semantically meaningful and diverse item sequences. This creates a fundamental disconnect between training and deployment objectives.

2. Probability Distribution Skew: Common tokens (e.g., articles, conjunctions) accumulate disproportionately high probabilities during training, despite contributing minimal semantic value to item differentiation. This skew manifests itself in beam search paths converging prematurely through these high-probability but low-information tokens.

3. Constrained Exploration: The multiplicative nature of sequence probabilities, combined with beam width constraints, creates a strong bias toward paths beginning with common tokens. This systematically limits the model's ability to explore diverse recommendation candidates, even when such alternatives might better serve the user's interests.

4 Methodology

4.1 Decision Modeling

To measure the decisiveness of tokens, We propose modeling the LLM generation process as a sequential decision-making task, where each token generation step represents a decision that progressively narrows the space of possible recommendations. By introducing a reference item language set, we can quantitatively measure how each generated token contributes to determining the final recommendation.

Let $\mathcal{I}$ denote the complete set of available items, and $\mathcal{I}^{y_{\leq t}}$ represent the subset of items consistent with the prefix sequence $y_{\leq t}$ generated up to step t. The reference set allows us to track how the space of possible recommendations narrows with each token and measure the uncertainty reduction contributed by each decision.

To quantify the uncertainty at each generation step, we employ **Shannon entropy** over the reference item distribution:

$$H(y_{\leq t}) = -\sum_{\mathcal{I}_i \in \mathcal{I}^{y \leq t}} p_r(\mathcal{I}_i) \log p_r(\mathcal{I}_i) \tag{5}$$

where $p_r(\mathcal{I}_i)$ represents the probability that the item $\mathcal{I}_i$ is the correct recommendation given the generated sequence $y \leq t$. Higher entropy values indicate greater uncertainty about the final recommendation.

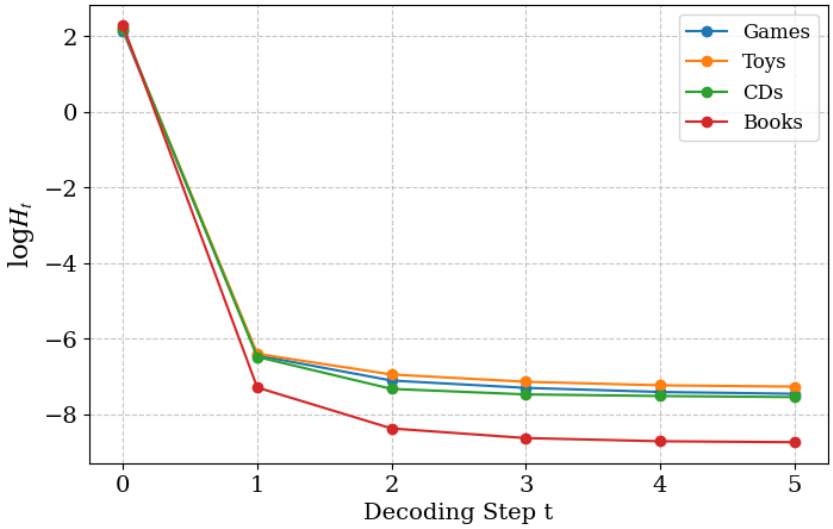

Figure 1: Entropy change with decoding steps.

The central metric for evaluating token decisiveness is **Information Gain (IG)**, which measures how much a token reduces uncertainty about the final recommendation:

$$\mathrm{IG}(y_t|y_{<t-1}) = H(y_{<t-1}) - H(y_{<t}) \tag{6}$$

This formulation directly quantifies the contribution of token y_t to narrowing the recommendation space. Tokens with high IG provide substantial information for determining the final recommendation, while tokens with low IG contribute minimally to the decision process despite potentially having high generation probabilities.

4.2 Token-level Bias Analysis

Leveraging our decision modeling framework, we conduct experiments to analyze how tokens with different information gain values affect model behavior during both fine-tuning and decoding phases. These analyses reveal systematic biases in how LLMs learn and generate tokens based on their decisiveness.

4.2.1 Bias in SFT

We monitor loss convergence throughout the fine-tuning process by grouping tokens into three categories based on their information gain:

Zero IG tokens: Tokens that do not provide discriminatory information (about 70%)

Our experiments demonstrate that the loss for high IG tokens remains consistently elevated throughout training, while zero IG tokens rapidly converge to low loss values. This disparity indicates that tokens with high information gain are inherently more difficult for models to learn, while non-deterministic tokens are readily learned. Consequently, this leads to a non-uniform distribution across the item language space, where semantically crucial tokens are underrepresented in the model's learned distributions.

(How to explain the bias in SFT using token grouping?)

4.2.2 Bias in Decoding

To analyze decoding bias, we compare entropy distributions between model-generated recommendations and ground truth items. For each decoding step t, we collect the top 10

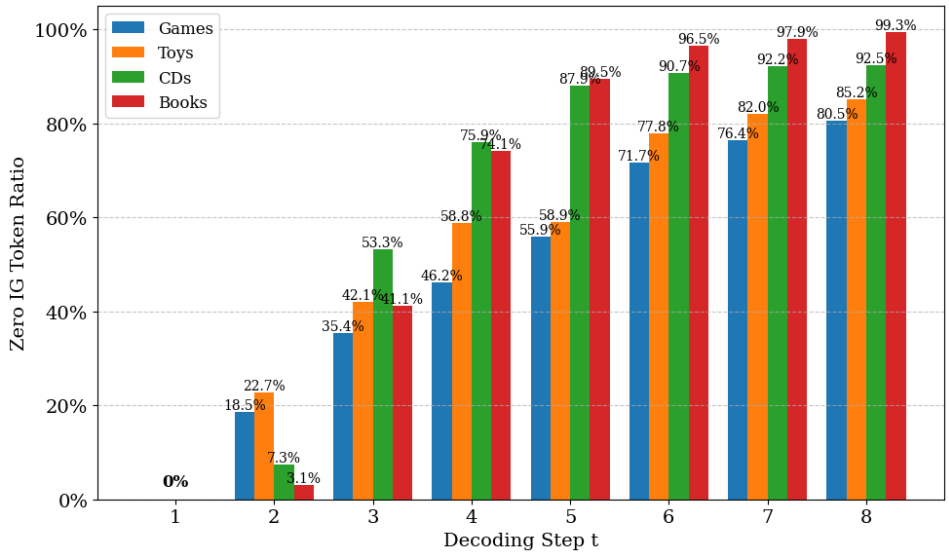

Figure 2: Entropy change with decoding steps.

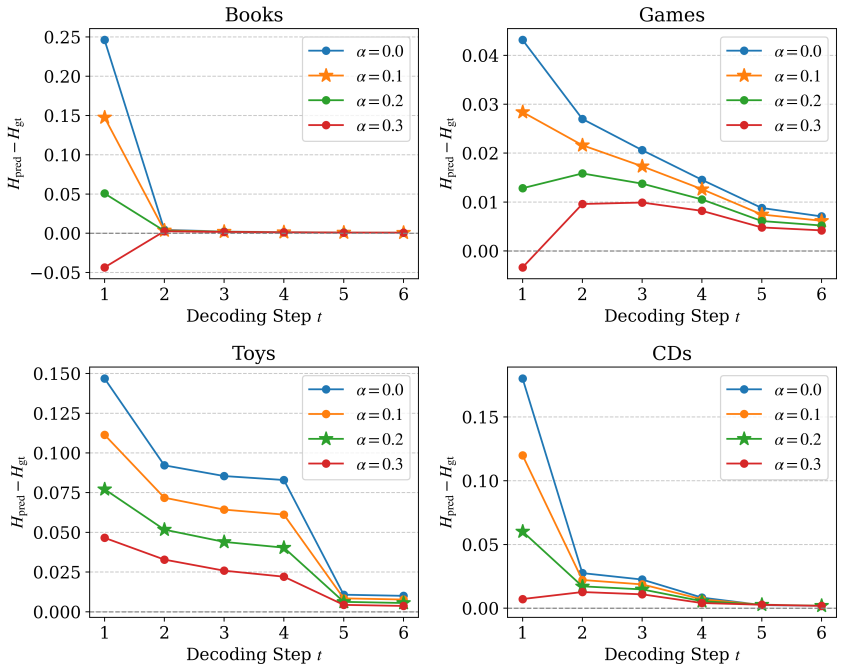

Figure 3: Entropy difference between Prediction and Ground Truth

items predicted by the LLM and their corresponding ground truth references. As shown in Figure 1, the average entropy of predicted items consistently exceeds that of ground truth items across all decoding steps. This systematic entropy gap reveals that **compared to the natural data distribution (ground truth), LLMs consistently prefer tokens with lower information gain during generation.**

Based on these experimental findings, we identify two critical biases in current LLM-based recommendation systems:

1. Learning Bias: During fine-tuning, the model exhibits a systematic bias toward learning non-decisive tokens more effectively than decisive ones. This learning disparity results in the "ghost token" phenomenon, where non-deterministic tokens receive disproportionately high logits despite contributing minimal semantic value.

2. Decoding Bias: During decoding, beam search tends to prematurely prune paths containing high information gain tokens due to their lower logit values. This pruning behavior manifests in the homogeneity issue, where multiple recommended items share common prefixes composed of low-information tokens, ultimately reducing recommendation diversity.

4.3 IG-based Debiasing

Based on our analysis of token-level biases, we propose a two-stage debiasing approach that leverages information gain to improve both model training and decoding. Our method addresses the identified biases by reweighting both the training loss and decoding scores according to token decisiveness.

4.3.1 Loss Weighting in SFT

To address the learning bias toward non-decisive tokens, we introduce a weighted training objective:

$$\mathcal{L}_D = \frac{1}{\Omega} \sum_{t=1}^{|y|} w_t l(f_\theta(x, y_{<t}); y_t), \tag{7}$$

where the weight w for each token is computed as:

$$w_t = \begin{cases} \beta, & \text{if } IG(y_t) = 0 \\ 1, & \text{if } IG(y_t) > 0 \end{cases} \tag{8}$$

The hyperparameter $\beta \in [0, 1]$ controls the strength of the weighting effect, with $\beta = 1$ reducing to the standard unweighted training objective. This formulation ensures that tokens with higher IG receive proportionally more attention during training.

4.3.2 Calibration in Decoding

To counteract the decoding bias against high-information tokens, we modify the beam search scoring function:

$$S(y_{\leq t}) = S(y_{\leq t-1}) + w_d \log p(y_t | x, y_{\leq t-1})) \tag{9}$$

$$w_d = 1 - \alpha \widetilde{IG}(y_t) \tag{10}$$

5 Experiments

5.1 Efficiency Analysis

Papers to be submitted to COLM 2025 must be prepared according to the instructions presented here.

Performance Analysis: Fine-tuning CFT:

Authors are required to use the COLM LaTeX style files obtainable at the COLM website. Please make sure you use the current files and not previous versions. Tweaking the style files may be grounds for rejection.

Table 1: Performance comparison across different datasets and models

Method	**Books**				**Games**			
	N@5	N@10	H@5	H@10	N@5	N@10	H@5	H@10
GRU4Rec	0.0060	0.0078	0.0094	0.0149	0.0169	0.0221	0.0261	0.0423
SASRec	0.0097	0.0123	0.0146	0.0226	0.0237	0.0290	0.0338	0.0502
BIGRec	0.0190	0.0211	0.0245	0.0309	0.0317	0.0381	0.0430	0.0631
+Pos	0.0197	0.0218	0.0255	0.0319	0.0319	0.0396	0.0423	0.0665
+CFT	0.0195	0.0218	0.0250	0.0321	0.0349	0.0414	0.0482	0.0686
+IGD	0.0267	0.0294	0.0334	0.0419	0.0423	0.0507	0.0576	0.0833
Improvement	*+41.3%*	*+40.0%*	*+36.9%*	*+36.0%*	*+33.4%*	*+33.1%*	*+34.0%*	*+32.0%*
D3	0.0212	0.0228	0.0266	0.0315	0.0415	0.0477	0.0581	0.0773
+Pos	0.0221	0.0237	0.0275	0.0324	0.0429	0.0489	0.0581	0.0767
+CFT	0.0219	0.0236	0.0275	0.0327	0.0437	0.0499	0.0613	0.0806
+IGD	0.0291	0.0313	0.0356	0.0424	0.0518	0.0598	0.0705	0.0946
Improvement	*+37.3%*	*+37.3%*	*+33.8%*	*+34.6%*	*+25.6%*	*+29.2%*	*+26.7%*	*+22.7%*

Method	**Toys**				**CDs**			
	N@5	N@10	H@5	H@10	N@5	N@10	H@5	H@10
GRU4Rec	0.0200	0.0238	0.0275	0.0392	0.0248	0.0288	0.0342	0.0467
SASRec	0.0356	0.0398	0.0473	0.0745	0.0477	0.0535	0.0647	0.0824
BIGRec	0.0553	0.0623	0.0736	0.0951	0.0502	0.0553	0.0623	0.0782
+Pos	0.0561	0.0631	0.0741	0.0958	0.0511	0.0566	0.0632	0.0802
+CFT	0.0561	0.0630	0.0746	0.0961	0.0509	0.0566	0.0631	0.0810
+IGD	0.0577	0.0656	0.0771	0.1014	0.0540	0.0593	0.0669	0.0833
Improvement	*+4.34%*	*+5.30%*	*+4.76%*	*+6.62%*	*+7.78%*	*+7.82%*	*+9.33%*	*+9.04%*
D3	0.0634	0.0698	0.0833	0.1029	0.0716	0.0767	0.0882	0.1040
+Pos	0.0644	0.0702	0.0850	0.1029	0.0729	0.0779	0.0902	0.1053
+CFT	0.0640	0.0704	0.0840	0.1036	0.0736	0.0786	0.0917	0.1069
+IGD	0.0658	0.0726	0.0868	0.1082	0.0748	0.0801	0.0929	0.1092
Improvement	*+3.79%*	*+4.01%*	*+4.20%*	*+5.15%*	*+4.47%*	*+4.43%*	*+5.33%*	*+5.00%*

	Books				Games			
Methods	N@5	N@10	H@5	H@10	N@5	N@10	H@5	H@10
IGD	0.0291	0.0313	0.0356	0.0424	0.0520	0.0610	0.0711	0.0991
w/o w_d	0.0290	0.0312	0.0355	0.0422	0.0515	0.0593	0.0700	0.0938
w/o w_t	0.0212	0.0229	0.0268	0.0318	0.0414	0.0484	0.0575	0.0790
w/o both	0.0212	0.0227	0.0266	0.0314	0.0415	0.0477	0.0581	0.0773
	Toys				CDs			
Methods	N@5	N@10	H@5	H@10	N@5	N@10	H@5	H@10
IGD	0.0658	0.0726	0.0868	0.1082	0.0748	0.0801	0.0929	0.1092
w/o w_d	0.0653	0.0719	0.0861	0.1063	0.0751	0.0800	0.0926	0.1077
w/o w_t	0.0640	0.0711	0.0843	0.1060	0.0718	0.0768	0.0887	0.1041
w/o both	0.0634	0.0698	0.0833	0.1029	0.0716	0.0767	0.0882	0.1040

Table 2: Ablation

5.1.1 Copy Options

If your paper is ultimately accepted, the option `\final` should be set for the `\usepackage[submission]{colm2025_conference}` command for the camera ready version. The `submission` options is the default, and is to be used for all submissions during the review process. It also turns on the line numbers. If you wish to submit a preprint, the option `preprint` should be used.

Method	Training Time
Default	d
Pos	$1.0275 \times d$
CFT	$2.0529 \times d$
IGD	$1.0354 \times d$

Table 3: Comparison of training time for different methods relative to the default trainer.

Table 4: Dataset Statistics and Token Information Gain Analysis

Dataset	Items	Train	Valid	Test	Tokens	Zero IG Tokens (%)
Books	41,722	682,998	85,376	85,376	7,183,839	5,241,997 (72.97%)
Games	11,037	201,613	25,202	25,203	2,128,430	1,292,171 (60.71%)
Toys	11,252	112,755	14,095	14,096	1,530,370	1,098,070 (71.75%)
CDs	14,239	148,685	18,586	18,587	805,786	450,960 (55.96%)

References

Keqin Bao, Jizhi Zhang, Wenjie Wang, Yang Zhang, Zhengyi Yang, Yancheng 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*, 2023a. ISSN 2770-6699.

Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 1007–1014, 2023b.

Keqin Bao, Jizhi Zhang, Yang Zhang, Xinyue Huo, Chong Chen, and Fuli Feng. Decoding matters: Addressing amplification bias and homogeneity issue for llm-based recommendation. *arXiv preprint arXiv:2406.14900*, 2024.

Chongming Gao, Ruijun Chen, Shuai Yuan, Kexin Huang, Yuanqing Yu, and Xiangnan He. Sprec: Leveraging self-play to debias preference alignment for large language model-based recommendations. *arXiv preprint arXiv:2412.09243*, 2024.

Chongming Gao, Mengyao Gao, Chenxiao Fan, Shuai Yuan, Wentao Shi, and Xiangnan He. Process-supervised llm recommenders via flow-guided tuning. *arXiv preprint arXiv:2503.07377*, 2025.

Jesse Harte, Wouter Zorgdrager, Panos Louridas, Asterios Katsifodimos, Dietmar Jannach, and Marios Fragkoulis. Leveraging large language models for sequential recommendation, 2023. URL https://doi.org/10.1145/3604915.3610639.

Jiayi Liao, Sihang Li, Zhengyi Yang, Jiancan Wu, Yancheng Yuan, Xiang Wang, and Xiangnan He. Llara: Large language-recommendation assistant. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1785–1795, 2024.

Jianghao Lin, Xinyi Dai, Yunjia Xi, Weiwen Liu, Bo Chen, Hao Zhang, Yong Liu, Chuhan Wu, Xiangyang Li, Chenxu Zhu, et al. How can recommender systems benefit from large language models: A survey. *ACM Transactions on Information Systems*, 43(2):1–47, 2025.

Xinyu Lin, Wenjie Wang, Yongqi Li, Fuli Feng, See-Kiong Ng, and Tat-Seng Chua. Bridging items and language: A transition paradigm for large language model-based recommendation, 2024a. URL https://doi.org/10.1145/3637528.3671884.

Xinyu Lin, Wenjie Wang, Yongqi Li, Shuo Yang, Fuli Feng, Yinwei Wei, and Tat-Seng Chua. Data-efficient fine-tuning for llm-based recommendation, 2024b. URL https://doi.org/10.1145/3626772.3657807.

Qijiong Liu, Nuo Chen, Tetsuya Sakai, and Xiao-Ming Wu. Once: Boosting content-based recommendation with both open- and closed-source large language models. *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, 2024. URL https://api.semanticscholar.org/CorpusID:258615357.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, and Qi Liu. A survey on large language models for recommendation. *World Wide Web*, 27(5):60, 2024. ISSN 1386-145X.

Zhenrui Yue, Sara Rabhi, Gabriel de Souza Pereira Moreira, Dong Wang, and Even Oldridge. Llamarec: Two-stage recommendation using large language models for ranking. *arXiv preprint arXiv:2311.02089*, 2023.

Junjie Zhang, Ruobing Xie, Yupeng Hou, Xin Zhao, Leyu Lin, and Ji-Rong Wen. Recommendation as instruction following: A large language model empowered recommendation approach. *ACM Transactions on Information Systems*, 2023.

Yang Zhang, Juntao You, Yimeng Bai, Jizhi Zhang, Keqin Bao, Wenjie Wang, and Tat-Seng Chua. Causality-enhanced behavior sequence modeling in llms for personalized recommendation. *arXiv preprint arXiv:2410.22809*, 2024.

Zihuai Zhao, Wenqi Fan, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Zhen Wen, Fei Wang, Xiangyu Zhao, Jiliang Tang, et al. Recommender systems in the era of large language models (llms). *IEEE Transactions on Knowledge and Data Engineering*, 2024.

A Appendix

You may include other additional sections here.