Does LLM Focus on the Right Words? Mitigating Context Bias in LLM-based Recommenders

Bohao Wang[†‡]
bohao.wang@zju.edu.cn
Zhejiang University
Hangzhou, China

Jiawei Chen[*†‡§]
sleepyhunt@zju.edu.cn
Zhejiang University
Hangzhou, China

Feng Liu
liufeng4hit@gmail.com
OPPO Research Institute
Shenzhen, China

Changwang Zhang
changwangzhang@foxmail.com
OPPO Research Institute
Shenzhen, China

Jun Wang
junwang.lu@gmail.com
OPPO Research Institute
Shenzhen, China

Canghong Jin
jinch@zucc.edu.cn
Hangzhou City University
Hangzhou, China

Chun Chen[†‡]
chenc@cs.zju.edu.cn
Zhejiang University
Hangzhou, China

Can Wang[†§]
wcan@zju.edu.cn
Zhejiang University
Hangzhou, China

Abstract

Large language models (LLMs), owing to their extensive open-domain knowledge and semantic reasoning capabilities, have been increasingly integrated into recommender systems (RS). However, a substantial gap remains between the pre-training objectives of LLMs and the specific requirements of recommendation tasks. To address this gap, supervised fine-tuning (SFT) is commonly performed on specially curated recommendation datasets to further enhance their predictive ability. Despite its success, SFT exhibits a critical limitation: it induces **Context Bias**, whereby the model over-relies on auxiliary tokens—such as task descriptions and prefix-generated tokens—while underutilizing core user interaction tokens that encode user-specific preferences. This bias not only undermines recommendation accuracy but also raises unfairness concerns.

To address this issue, we propose **Group Distributionally Robust Optimization-based Tuning (GDRT)**, a novel fine-tuning paradigm that enforces consistent model performance across token groups with varying degrees of relevance to auxiliary tokens. By adaptively upweighting underperforming groups, typically those weakly correlated with auxiliary tokens, GDRT shifts the model's attention from superficial auxiliary cues to informative user interaction tokens, thereby mitigating context bias. Extensive experiments conducted on three public datasets demonstrate that GDRT effectively mitigates context bias, yielding substantial improvements in recommendation accuracy (with an average NDCG@10 gain of 24.29%) and significantly enhancing recommendation fairness. The code is available at https://github.com/WANGBohaO-jpg/GDRT.

CCS Concepts

• **Information systems → Recommender systems**.

Keywords

Sequential Recommendation; Large Language Model; Bias

ACM Reference Format:
Bohao Wang, Jiawei Chen, Feng Liu, Changwang Zhang, Jun Wang, Canghong Jin, Chun Chen, and Can Wang. 2026. Does LLM Focus on the Right Words? Mitigating Context Bias in LLM-based Recommenders. In *Proceedings of the ACM Web Conference 2026 (WWW '26), April 13–17, 2026, Dubai, United Arab Emirates.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3774904.3792607

1 Introduction

With remarkable open-domain knowledge and semantic reasoning capabilities [1, 23], Large Language Models (LLMs) have been extensively explored for integration into recommendation systems (RS) [68]. One prominent approach involves positioning LLMs as the central recommendation backbone [2, 4, 32, 35, 38, 52, 58, 60, 61, 79, 81]. These methods express items as textual descriptions (*e.g.,* titles), and construct language prompts based on users' past interactions, which are then used to instruct LLMs to predict users' future interactions. Figure 1 illustrates the mechanism of such LLM-based recommendation. LLMs operate at a fine-grained semantic token level, sequentially generating tokens of the predicted items by analyzing their nuanced semantic relations with the user's previous interactions and other auxiliary information (*e.g.,* task descriptions, prefix tokens of the predicted item). This fine-grained paradigm enables the capture of subtle semantic patterns in user interests and thus represents a promising direction for advancing recommender systems [2].

To better align LLMs with recommendation objectives and capture collaborative filtering signals, Supervised Fine-Tuning (SFT)

[*] Corresponding author.
[†] State Key Laboratory of Blockchain and Data Security, Zhejiang University.
[‡] College of Computer Science and Technology, Zhejiang University.
[§] Hangzhou High-Tech Zone (Binjiang) Institute of Blockchain and Data Security.

WWW '26, Dubai, United Arab Emirates
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2307-0/2026/04
https://doi.org/10.1145/3774904.3792607

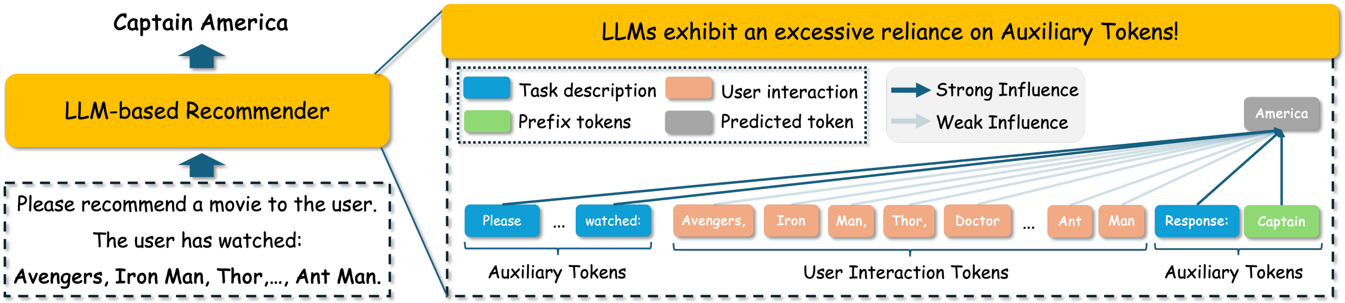

Figure 1: Illustration of LLM-based recommendations and context bias, wherein the model exhibits an over-reliance on auxiliary tokens (*i.e.,* Task description, Prefix tokens) and insufficient utilization of User interaction during generation.

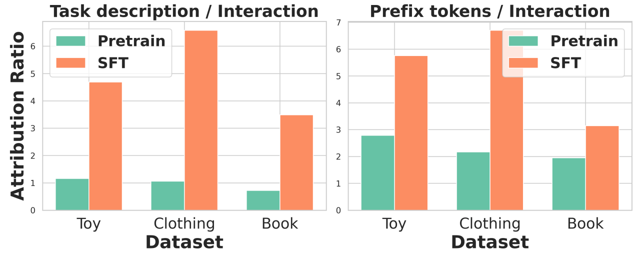

Figure 2: Ratio of attribution values between auxiliary tokens and user interaction tokens before and after SFT. Left: task description vs. user interaction tokens. Right: prefix tokens of predicted item (take the first token) vs. user interaction tokens.

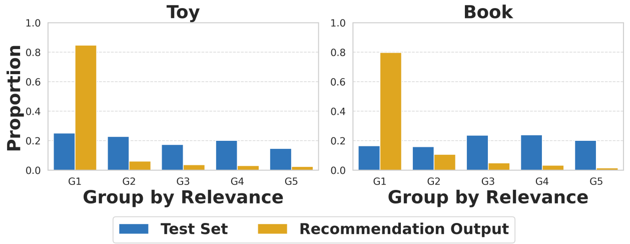

Figure 3: Distribution of Top-1 recommended items generated by the SFT-trained model across five group defined according to the semantic relevance of items to the auxiliary tokens (Group 1: highest relevance, Group 5: lowest relevance). We also present the distribution of the test set across the same five groups for comparison.

is commonly employed [2, 4, 38, 58]. In this strategy, each prompt is paired with the target item description, and the LLM is fine-tuned to generate the correct prediction. This procedure enables the model to capture semantic relationships between prompts and targets present in the training data, often resulting in substantial performance gains.

However, we find that SFT introduces a significant **Context Bias**. Specifically, SFT drives the model to over-rely on auxiliary tokens (*e.g.,* task descriptions or prefix tokens) while under-utilizing core interaction tokens that encode user personalized preferences as shown in Figure 1. To verify this, we conduct Feature Ablation Attribution analysis [33, 44], a standard approach to quantify each token's contribution to model predictions. Figure 2 shows that SFT dramatically amplifies the relative impact of auxiliary tokens while significantly suppressing the influence of interaction tokens, with the influence ratio shifting from about 1:1 before fine-tuning to more than 6:1 afterward as measured on typical Amazon datasets.

This over-reliance reveals shortcut learning: LLMs simply memorize correlations with frequently occurring auxiliary tokens rather than grounding predictions in user-specific preferences. Such bias not only undermines recommendation accuracy but also raises serious fairness concerns. Specifically, this bias skews recommendations toward a narrow subset of items whose tokens exhibit higher semantic relevance to auxiliary tokens. To verify this, we divided items into five groups based on their relevance to auxiliary tokens. As shown in Figure 3, over 80% of recommended items fall into the highest relevance group, while this group comprises only 20% of target items in the test set. These findings motivate our core research question: **How can we mitigate context bias in LLM-based recommenders?**

To address this, we propose a novel fine-tuning strategy **GDRT**, that leverages Group Distributionally Robust Optimization (Group DRO) [50] to mitigate context bias. We first group training instances according to the semantic relevance between the target token and the auxiliary tokens, which can be evaluated by the LLM's predictive probability when user history is masked. Group DRO then enforces LLMs to perform consistently well across all groups, regardless of their relevance strength with auxiliary tokens. This optimization objective naturally shifts the LLM's attention away from auxiliary tokens toward user-specific interaction tokens, as simple reliance on shortcut auxiliary tokens results in poor performance on groups with weaker correlations. Importantly, GDRT is easy to implement and computationally efficient, requiring only high-efficiency group construction and dynamic sample weighting during training. It can be seamlessly integrated into various LLM-based recommenders, yielding improvements in both accuracy and fairness.

Our main contributions are:

- We provide a comprehensive empirical analysis revealing that SFT in LLM-based recommendation induces significant context bias, negatively affecting both accuracy and fairness.
- We propose GDRT, a Group DRO-based fine-tuning strategy, to effectively mitigate context bias in LLM-based recommendation.
- We conduct extensive experiments demonstrating that GDRT achieves state-of-the-art recommendation performance in both accuracy and fairness metrics.

2 Preliminary

2.1 LLM-based Recommendation

Following previous work [2, 4, 38, 39, 45, 78], this paper also focuses on sequential recommendation [30], a conventional recommendation scenario in practice. Let $\mathcal{V}$ denote the set of items in the recommendation system. Given a user's historical interaction sequence $S = \{s_1, s_2, ..., s_n\}$, where each $s_i \in \mathcal{V}$ represents the i-th interacted item, the goal of the RS is to predict the user's next interaction s_{n+1} that the user is likely to interact with.

The remarkable success of LLMs across diverse domains [8, 11, 47, 54, 62, 66] has spurred growing interest in their application to recommendation systems (RS) [68]. A prominent approach is to directly leverage LLMs as recommenders [35]. As shown in Figure 1, this paradigm constructs a language prompt $x = [\, x^{\text{task}}; x^{\text{user}} \,]$, where x^{task} represents the task description and x^{user} denotes the textual form (*e.g.,* titles) of a user's historical interactions. This prompt then guides the LLM to generate the descriptions of recommended items y. Notably, LLMs operate at a fine-grained semantic token level, sequentially generating tokens of the predicted items according to the model estimated probability $P_\theta(y_t|x, y_{<t})$, where y_t denotes the t-th predictive tokens and $y_{<t}$ denotes the prefix tokens of the prediction. This fine-grained token-level paradigm has the potential to capture subtle semantic patterns in user preferences.

To align LLMs with recommendation objectives, supervised fine-tuning (SFT) is commonly applied, fine-tuning all or part of the model parameters using recommendation data [2, 4, 38, 81]. In this process, the training data is reorganized into a set of prompt–target pairs $\mathcal{D} = \{(x_i, y_i^*)\}_{i=1}^N$ where each x_i is the constructed prompt and y_i^* is the textual description of the target item. The LLM is optimized with the following log-likelihood objective:

$$\mathcal{L}_{SFT} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{|y_i^*|}\log P_\theta\left(y_{i,t}^* \mid x_i, y_{i,<t}^*\right) \tag{1}$$

where $|y_i^*|$ denotes the token length of the target item description. SFT increases the generative probability of the target item, encouraging the LLM to capture the inherent token-level semantic correlations between each target token $y_{i,t}^*$ and the user interactions x_i^{user}, task descriptions x^{task} and the prefix tokens $y_{i,<t}^*$. This process often yields substantial performance improvements [4].

2.2 Analyses on Context Bias

In this section, we first identify the context bias in fine-tuning LLM for recommendation, followed by discussing its negative effect. We then analyze the underlying causes of this bias and discuss why existing methods can not effectively address this issue. [1]

2.2.1 Empirical Evidence Demonstrating Context Bias. We conduct Feature Ablation Attribution (FAA) [33, 44] analysis to quantify contribution of different token types to the model predictions. FAA is an attribution method used to evaluate the importance of individual input components by measuring how the model's output changes when specific inputs are masked, and it has been widely adopted in the LLMs for interpreting token-level contributions [5, 76, 80]. A higher attribution value indicates a greater influence of input tokens on the model's output. Figure 2 shows the ratio of attribution values between auxiliary tokens (*e.g.,* task descriptions or prefix tokens) and user-specific interaction tokens. We report this ratio both before and after fine-tuning to enable direct comparison. From these results, we make the following observation:

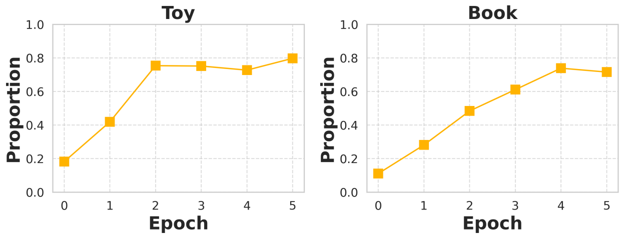

Figure 4: Proportion of Top-1 recommendations belonging to Item Group 1 (highest relevance with auxiliary tokens) over the course of SFT.

Context Bias: *Supervised fine-tuning (SFT) can bias LLMs to over-rely on auxiliary tokens while under-utilizing core interaction tokens that encode user personalized preferences.*

Before fine-tuning, the attribution value ratio between task description and interaction tokens is approximately 1:1. After fine-tuning, this ratio is markedly amplified across all datasets (*e.g.,* Toy: 4.69:1; Clothing: 6.58:1). A similar phenomenon is observed when comparing prefix tokens with interaction tokens. These results clearly indicate the presence of context bias, whereby the model disproportionately relies on auxiliary tokens rather than more informative interaction tokens.

2.2.2 Negative Effect of Context Bias. Context bias can substantially hinder the effectiveness of LLM-based recommenders, leading not only to decreased recommendation accuracy but also to pronounced unfairness issues. On the one hand, critical user–item interaction signals that capture user preferences may be ignored by the model, severely impairing its ability to deliver personalized recommendations. On the other hand, the recommendation output becomes inherently skewed toward a limited subset of items whose textual tokens exhibit strong correlations with auxiliary tokens.

To empirically verify this phenomenon, we conduct experiments on the typical Amazon datasets. Specifically, we partition items into five groups based on their semantic relevance to auxiliary tokens, measured by the model's estimated probability:

$$r(y) = \frac{1}{|y|}\sum_{t=1}^{|y|}\log P_\theta(y_t|x^{\text{task}}, y_{<t}) \tag{2}$$

Here we mask the user historical information in the original prompt template with the placeholder 'N/A', and compute the probability of the tokens for each item. This measure serves as an indicator of the relevance between the input and output, a practice commonly adopted for estimating semantic relevance [31]. Based on this metric, items are sorted and evenly divided into five groups, with Group 1 containing the items of highest relevance and Group 5 the least.

Next, we compute the proportion of Top-1 recommended items from each group, with the results shown in Figure 3. The analysis

[1] The experimental configuration in this section is consistent with our main experimental setup described in Section 4.1.3. We also provide additional analyses on other Prompt templates and LLMs in Appendix A.1.

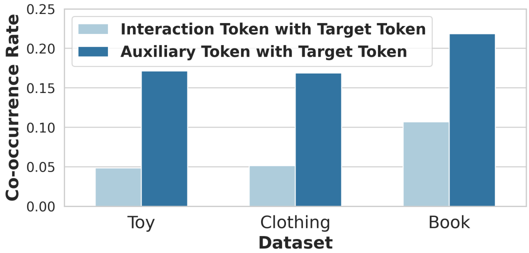

Figure 5: The co-occurrence rate of different types of token pairs in the training set.

reveals that fine-tuned LLMs display a pronounced inclination for items in Group 1, whose tokens exhibit the highest relevance to auxiliary tokens. Nearly 80% of Top-1 recommendations fall within this group, despite such items comprising only about 20% of the test set. As illustrated in Figure 4, this bias becomes progressively more pronounced as the tuning proceeds. This bias not only undermines recommendation accuracy but also exacerbates exposure unfairness by systematically over-promoting items misaligned with individual user preferences. Such skew can severely degrade user experience and distort the recommendation ecosystem. For example, it would incentivize content providers to adopt clickbait-like titles or other superficial textual strategies to artificially increase token relevance to the auxiliary tokens.

2.2.3 Origins of Context Bias. The emergence of context bias can be traced to biases inherent in the training data. As shown in Figure 5, the co-occurrence rate between auxiliary tokens and target item tokens is significantly higher than those between interaction tokens and target item tokens. This is expected, as task prompts appear in every training instance, and prefix tokens are always accompanied for item tokens. Consequently, during fine-tuning, the LLM tends to capture shortcut patterns, simply memorizing frequent correlations with auxiliary tokens, while neglecting the more important user-specific interaction signals. This incurs context bias and motivates the need for improved fine-tuning strategies.

2.2.4 Limitations of Related Strategies. Several recent studies have explored bias and fairness issues in LLM-based RS. However, the characteristics of these biases differ from those of *context bias*. Specifically, context bias arises during fine-tuning and reflects the inherent over-reliance of LLMs on auxiliary tokens. In contrast, earlier work has primarily focused on biases such as *popularity bias* [21, 22, 29, 37, 41], arising from imbalanced item frequency in training data; *position bias* [6, 7, 17, 27, 28, 42, 43, 72], caused by the model’s sensitivity to the ordering of candidate items; *amplification bias* [3], arising from length normalization, which favors items containing tokens with generation probabilities close to 1. These biases stem from distinct sources and mechanisms. Consequently, this newly identified form of context bias warrants explicit and targeted mitigation, yet recent work has fallen short in addressing this issue (*cf.* Table 2).

Another related work, CFT [75], aims to encourage LLMs to better leverage user interaction information. CFT introduces counterfactual learning to forcibly enhance the influence of interaction tokens for each training instance. However, this strategy suffers from multiple aspects of limitations: **(1) Objective misalignment**. The counterfactual objective is not directly aligned with improving recommendation accuracy or fairness, and its contribution to these aspects remains uncertain. In fact, CFT only treats this objective as an auxiliary loss and over-emphasis on this term has been observed to lead to substantial performance degradation. **(2) Difficulty in weight selection**. CFT relies on manually specified weights to determine the degree to which a training instance should rely on interaction tokens. In fact, the optimal weights can vary substantially across instances and are difficult to estimate accurately. Manual specification is therefore challenging, prone to deviation from the ideal value, and ultimately detrimental to model effectiveness. Empirically, even when we directly employ the official source code of CFT and conduct a fine-grained hyperparameter search, CFT yields only limited performance gains on some datasets (*cf.* Table 2). **(3) High computational overhead**. CFT requires to process counterfactual instances, resulting in more than double the training time compared with SFT (*cf.* Figure 12). In contrast, our GDRT is explicitly aligned with the target objective, enabling the model to consistently perform well across different instance groups. This naturally encourages greater focus on interaction tokens without requiring manual specification of influence strengths.

3 Methodology

The above analyses reveal a significant context bias inherent in fine-tuning LLMs for recommendation. To address this issue, we propose a novel fine-tuning framework, termed GDRT, which leverages Group Distributionally Robust Optimization (Group DRO) [50] to mitigate such bias. This section first outlines the general idea of GDRT, and then describes the group partitioning strategy and customized loss function.

General Idea. Rather than directly intervening the learning process to manually enhance the effect of interaction tokens, we pursue an alternative objective: *ensuring that the model achieves consistently strong performance across target tokens regardless of their degree of relevance to auxiliary tokens.* It naturally shifts the LLM’s attention away from auxiliary tokens towards user-specific interaction tokens, as simple reliance on shortcut auxiliary tokens results in poor performance on the instances with weak correlations. Besides, this objective is directly aligned with the goals of high recommendation accuracy and fairness.

To implement this idea, we adopt Group DRO, which partitions the training data into multiple groups and uses an adversarial training mechanism to encourage great performance across all groups. Group DRO has been widely applied in various domains and shown effectiveness in mitigating group disparities and shortcut correlations [46, 50, 77]. In applying Group DRO to our problem, we address two key questions: (1) how to construct groups that capture varying degrees of relevance to auxiliary tokens; and (2) how to design a loss function that is computationally efficient.

Token Grouping by Relevance with Auxiliary Tokens. Our objective is to ensure consistent model performance across groups that differ in their degree of relevance to auxiliary tokens. Accordingly, the constructed groupings should capture variations in this relevance. Towards this end, we compute the predictive probability of each target token conditioned solely on the corresponding

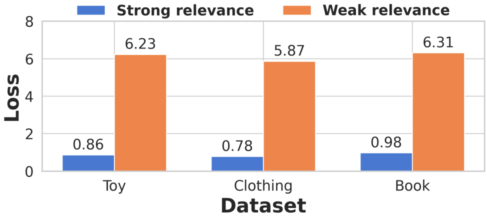

Figure 6: Loss of token groups with varying degrees of relevance to auxiliary tokens in the SFT-trained biased model. Target tokens are grouped into strong- and weak-relevance sets using K-means on relevance scores computed with Eq. 3.

auxiliary tokens:

$$r(y^*_{i,t}) = \log P_\theta\left(y^*_{i,t} \mid x^{\text{task}}, y^*_{i,<t}\right). \tag{3}$$

where $y^*_{i,t}$ denotes the t-th token of the target item in the i-th training sample, and $y^*_{i,<t}$ denotes its prefix tokens. Based on these relevance scores, all target tokens are partitioned into G disjoint groups $\{\mathcal{G}_1, \ldots, \mathcal{G}_G\}$ using the K-means algorithm[2], where G is a hyperparameter that determines the granularity of grouping. Each group $\mathcal{G}_g$ contains the index (i,t) of target tokens exhibiting similar degrees of relevance to auxiliary tokens, whereas tokens belonging to different groups demonstrate distinct relevance levels.

Unlike Eq. 2, which evaluates *item-wise* relevance between the entire target item and the auxiliary tokens, Eq. 3 computes *token-wise* relevance. This distinction is crucial, as different tokens within the same target item may exhibit varying degrees of relevance to the auxiliary tokens. Such finer-grained grouping can more effectively capture variations in this relevance.

The Objective of GDRT. Subsequently, we employ Group DRO to enforce consistent model performance across groups with different strength of relevance. Specifically, the training objective of GDRT is defined as:

$$\begin{aligned} \mathcal{L}_{GDRT} &= \max_{Q} \sum_{g=1}^{G} Q(g)\mathcal{L}(g) \quad \text{s.t.} \quad D_{KL}(Q,U) \le \eta \\ \mathcal{L}(g) &= \frac{1}{|\mathcal{G}_g|} \sum_{(i,t)\in\mathcal{G}_g} -\log P_\theta\left(y^*_{i,t} \mid x_i, y^*_{i,<t}\right) \end{aligned} \tag{4}$$

where $\mathcal{L}(g)$ represents the vanilla generative loss for group $\mathcal{G}_g$, and $|\mathcal{G}_g|$ represents the number of target tokens in group $\mathcal{G}_g$. The term Q denotes the weight distribution over groups, acting as a flexible adversarial perturbation to the original empirical distribution, with $Q(g)$ being the weight assigned to the g-th group. This weight distribution is regularized via a Kullback–Leibler divergence term $D_{KL}(Q,U)$ between the weight distribution Q and the uniform distribution U, with the parameter η controlling the perturbation magnitude.

Comparing $\mathcal{L}_{GDRT}$ (Eq. 4) with the original SFT objective $\mathcal{L}_{SFT}$ (Eq. 1), the main difference lies in the introduction of the additional group weighting term $Q(g)$. Intuitively, Group DRO imposes an adversarial shift to the group distribution by perturbing the relevance distribution with respect to auxiliary tokens, thereby compelling the model to perform consistently well across groups under different adversarial re-weightings. This mechanism naturally mitigates the model's reliance on auxiliary tokens, and encourages the model to perform consistently well across different groups.

Table 1: Statistics of the datasets.

Dataset	#Users	#Items	#Interactions	#Density
Toy	19124	11758	165247	0.0735%
Clothing	39230	22948	277534	0.0308%
Book	16559	6344	151928	0.1446%

Efficient Implementation. While Group DRO involves the complex adversarial optimization, it can be simplified to an equivalent closed-form objective. We have the following lemma:

Lemma 1. *Equation 4 can be reformulated as the following objective:*

$$\mathcal{L}_{GDRT} = \sum_{g=1}^{G} Q(g)\mathcal{L}(g), \quad Q(g) = \frac{e^{\mathcal{L}(g)/\tau}}{\sum_{g'=1}^{G} e^{\mathcal{L}(g')/\tau}} \tag{5}$$

The parameter τ is the dual Lagrange coefficient associated with the constraint $D_{KL}(Q,U) \le \eta$.

The lemma yields an explicit closed-form solution for the group weights $Q(g)$. From the perspective of these weights, the effect of Group DRO can be well understood: groups with higher losses, which typically correspond to tokens exhibit weaker relevance with auxiliary tokens (see Figure 6), receive larger weights. This adaptive reweighting encourages the model to allocate greater optimization emphasis to underperforming groups, thereby enhancing its learning on samples with low auxiliary-token relevance. As a result, the model attains more balanced performance across groups and effectively mitigates its over-reliance on auxiliary tokens.

This closed-form also facilitates an efficient implementation of GDRT. Compared to SFT, the additional steps just involve: partitioning the token groups based on Eq. 3, and dynamically updating the group weights according to Eq. 5. The time complexity of GDRT is the same as the basic SFT, and our empirical experiments also demonstrate their close running time (*cf.* Figure 12). Besides, this simple reformulation makes the integration of GDRT into existing LLM-based RS straightforward, requiring only minimal code changes.

4 Experiments

We aim to answer the following research questions:

- RQ1: How does GDRT perform compared to SOTA methods?
- RQ2: Can GDRT be integrated into other LLM-based RS?
- RQ3: Does GDRT mitigate context bias?
- RQ4: How do the hyperparameters affect the GDRT?
- RQ5: How does the efficiency of GDRT compare with baselines?

4.1 Experimental Settings

4.1.1 Datasets. Three widely used real-world datasets—*Amazon Toys and Games*, *Amazon Clothing, Shoes and Jewelry*, and *Amazon*

[2] We can simply employ the *scikit-learn* package to implement the K-means algorithm. Considering that $r(y^*_{i,t})$ is a numerical value, it can be evenly partitioned according to its magnitude, yielding comparable results.

Table 2: The performance comparison on three real-world datasets. The best result is bolded. Lower MGU and DGU indicate better fairness.

Dataset	Metric	SASRec	DROS	SASRec++	SFT	Reweight	D3	SPRec	CFT	GDRT
Toy	NDCG@5 ↑	0.0057	0.0095	0.0120	0.0118	0.0084	0.0115	0.0148	0.0119	**0.0152**
	NDCG@10 ↑	0.0073	0.0118	0.0144	0.0158	0.0121	0.0160	0.0175	0.0158	**0.0203**
	HIT@5 ↑	0.0090	0.0167	0.0200	0.0202	0.0146	0.0186	0.0234	0.0188	**0.0246**
	HIT@10 ↑	0.0138	0.0240	0.0271	0.0330	0.0261	0.0325	0.0317	0.0311	**0.0405**
	MGU@5 ↓	/	/	/	0.1870	0.2119	0.1740	0.2255	0.1964	**0.1288**
	DGU@5 ↓	/	/	/	0.6122	0.6655	0.5722	0.7197	0.6257	**0.4031**
Clothing	NDCG@5 ↑	0.0016	0.0043	0.0024	0.0038	0.0038	0.0049	0.0045	0.0042	**0.0054**
	NDCG@10 ↑	0.0024	0.0053	0.0033	0.0063	0.0066	0.0075	0.0072	0.0068	**0.0096**
	HIT@5 ↑	0.0032	0.0080	0.0042	0.0078	0.0074	0.0100	0.0084	0.0086	**0.0118**
	HIT@10 ↑	0.0058	0.0110	0.0070	0.0156	0.0160	0.0180	0.0168	0.0168	**0.0246**
	MGU@5 ↓	/	/	/	0.1553	0.2491	0.1211	0.1728	0.1410	**0.0522**
	DGU@5 ↓	/	/	/	0.4908	0.7988	0.4069	0.5443	0.4686	**0.1996**
Book	NDCG@5 ↑	0.0054	0.0060	0.0065	0.0067	0.0051	0.0073	0.0033	0.0080	**0.0139**
	NDCG@10 ↑	0.0071	0.0084	0.0081	0.0103	0.0070	0.0112	0.0071	0.0105	**0.0145**
	HIT@5 ↑	0.0089	0.0110	0.0100	0.0116	0.0100	0.0130	0.0057	0.0137	**0.0226**
	HIT@10 ↑	0.0141	0.0185	0.0153	0.0228	0.0160	0.0240	0.0178	0.0219	**0.0244**
	MGU@5 ↓	/	/	/	0.1433	0.1382	0.1229	0.1709	0.1657	**0.0579**
	DGU@5 ↓	/	/	/	0.4742	0.3406	0.4238	0.5477	0.4643	**0.2240**

Table 3: Performance comparison under different LLM-based RS. The best result is bolded.

Method	Toy		Clothing	
	NDCG@5	DGU@5	NDCG@5	DGU@5
MSL	0.0198	0.7162	0.0077	0.5815
MSL+GDRT	**0.0253**	**0.2697**	**0.0096**	**0.1469**
LLaRA	0.0131	0.6485	0.0041	0.5412
LLaRA+GDRT	**0.0168**	**0.4249**	**0.0053**	**0.2320**
A-LLM	0.0129	0.6122	0.0045	0.5238
A-LLM+GDRT	**0.0160**	**0.4192**	**0.0053**	**0.1817**

Books[3]—are employed in our experiments. To ensure fair comparison, we follow the data preprocessing procedures employed in recent literature [2, 16, 58]. Specifically, we first apply the 5-core setting to the raw datasets. For user interaction sequences exceeding 11 interactions, we segment the sequences using a sliding window of length 11. The segmented sequences are then sorted in ascending order by timestamp and partitioned into training, validation, and test sets in an 8:1:1 ratio. Due to the large size of *Amazon Books*, we randomly retain 100,000 items prior to 5-core processing. The statistics of the processed datasets are summarized in Table 1.

4.1.2 Baselines. The methods compared fall into several categories: **(1) Traditional RS**: SASRec [30], SASRec++ [34], DROS [71]. **(2) LLM-based RS**: SFT [2], CFT [75], MSL [58], LLaRA [38], A-LLM [32]. **(3) Debiasing for LLM-based RS**: Reweight [29], SPRec [21], D3 [3]. For a detailed description, see Appendix A.2.

4.1.3 Implementation Details. For all LLM-based methods, we adopt LLaMA3.2-3B [19] as the backbone, with the number of training epochs set to 5. The prompt design follows [2]. For inference, we consistently employ Constrained Beam Search (CBS) across all baselines, following prior work [3, 58], to ensure that the recommended items are drawn from the item set. The beam size is fixed at 10. For evaluation, we evaluate the NDCG@5 on the validation set for each epoch's checkpoint and select the checkpoint with the highest score for testing. The corresponding results on the test set are reported as the final performance. For the hyperparameters in GDRT, the number of groups G is tuned from $\{2, 5, 10\}$, and τ is tuned from $\{0.1, 0.2, 0.3, 0.5, 1.0\}$. To ensure fair comparisons, we utilize the source code provided by the original authors and tune the hyperparameters of all baseline methods according to the guidelines specified in their respective publications.

4.1.4 Metrics. Four widely used evaluation metrics are employed in this study: *NDCG@K* and *Hit Ratio@K* are used to evaluate recommendation accuracy, while *MGU@K* and *DGU@K* are adopted to evaluate fairness [29] (K=5, 10). Specifically, for the fairness metrics, items are divided into five groups according to Equation 2. We then calculate the discrepancy between each group's proportion in the Top-K recommendations and its proportion in the user's interaction history. *MGU@K* measures the average of these discrepancies, whereas *DGU@K* quantifies the gap between the maximum and minimum discrepancies across groups. Accordingly, smaller values of *MGU@K* and *DGU@K* indicate that the distribution of recommended items is more aligned with that of the user's historical interactions, reflecting better fairness.

4.2 Performance Comparison (RQ1 & RQ2)

Table 2 presents a comparative analysis of the proposed GDRT method against the baselines. Overall, GDRT demonstrates substantial performance improvements across all datasets. By mitigating context bias, reducing excessive reliance on auxiliary tokens, and enhancing the model's ability to capture user-specific behavioral patterns, GDRT delivers marked gains in both accuracy and fairness.

[3] https://jmcauley.ucsd.edu/data/amazon/index_2014.html

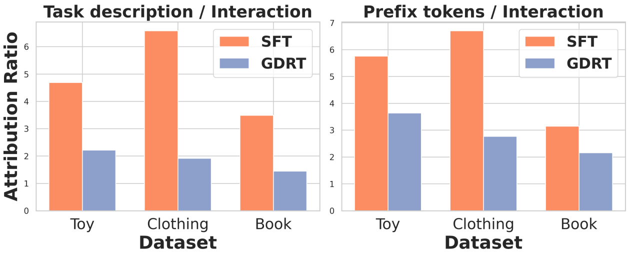

Figure 7: Ratio of attribution values between auxiliary tokens and user interaction tokens using SFT and GDRT. Left: task description vs. user interaction tokens. Right: prefix tokens of predicted item (take the first token) vs. user interaction tokens.

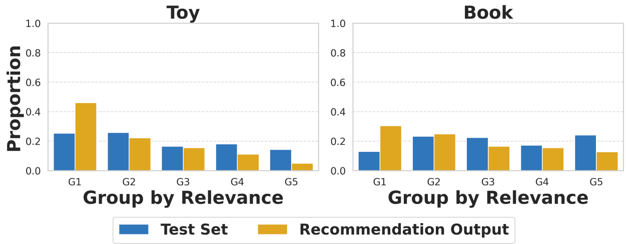

Figure 8: Distribution of Top-1 recommended items from the GDRT-trained model across five groups defined by auxiliary-token relevance (Group 1: highest relevance, Group 5: lowest relevance), with test-set distribution for comparison.

Specifically, GDRT achieves an average improvement of 24.29% in NDCG@10 and 37.43% reduction in MGU@5 compared with the best baseline. In contrast, other comparison methods yield only marginal improvements over SFT, and their lack of explicit mechanisms to address context bias results in limited fairness enhancement. Furthermore, since GDRT operates solely by modifying the loss function, it can be easily integrated into existing LLM-based RS without altering their architectures. To verify its general applicability, we further evaluate GDRT when incorporated into several advanced LLM-based recommendation methods. As shown in Table 3, GDRT consistently yields significant improvements in both performance and fairness across all evaluated methods, demonstrating its strong generalization capability. Additional comparisons using alternative prompt templates and LLMs beyond those used in the main experiments are provided in Appendix A.3.

4.3 In-depth Analysis (RQ3)

In this section, we present an empirical analysis of the effectiveness of GDRT in mitigating context bias. First, we perform FAA on both SFT- and GDRT-trained models, measuring the ratio of attribution values assigned to auxiliary tokens versus interaction tokens. As shown in Figure 7, GDRT yields a significantly lower ratio than SFT, indicating that GDRT effectively alleviates the model's over-reliance on auxiliary tokens. Furthermore, we analyze the recommendation distributions on different item groups with varying degrees of auxiliary-token relevance. As illustrated in Figure 8, the recommendation distribution generated by the GDRT-trained model closely align with the distribution of the test set. This improvement can be attributed to GDRT's ability to suppress the tendency of SFT to progressively amplify items highly correlated with auxiliary tokens as shown in Figure 9. Overall, these results demonstrate that GDRT effectively mitigates context bias.

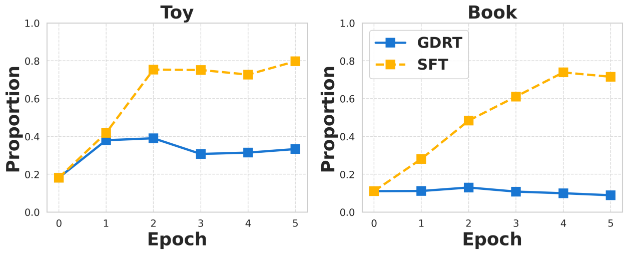

Figure 9: Proportion of Top-1 recommendations belonging to Item Group 1 (highest relevance with auxiliary tokens) during training with SFT and GDRT.

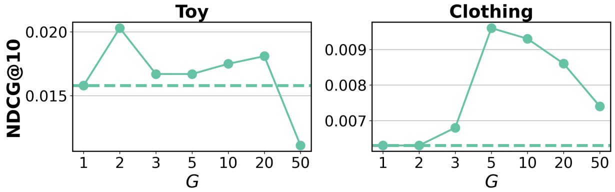

Figure 10: Hyperparameter sensitivity analysis on group number G (dashed: SFT baseline).

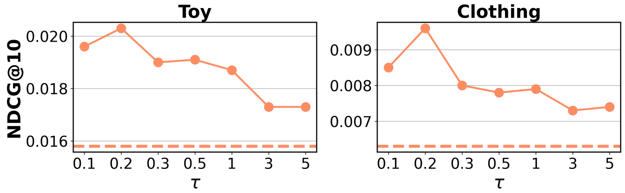

Figure 11: Hyperparameter sensitivity analysis on parameter τ in DRO (dashed: SFT baseline).

4.4 Hyper-Parameter Sensitivities (RQ4)

In this section, we perform a sensitivity analysis on the hyperparameters of GDRT: the number of groups G in K-means (Figure 10) and the coefficient τ in DRO (Figure 11). For both parameters, model performance exhibits a trend of initially increasing and then decreasing as the parameter values rise. This phenomenon can be explained as follows. For G, an excessively small value may group highly heterogeneous data together, thereby diminishing the differences in auxiliary-token relevance between groups. Conversely, an overly large G results in groups containing too few samples to reliably represent the underlying distribution. As for τ, a smaller value places greater emphasis on groups with higher losses; however, overemphasizing the worst-performing group can cause overfitting and impair overall generalization. In contrast, a larger τ balances optimization across all groups, but may reduce group-wise consistency in performance. Besides, the model demonstrates strong performance across a wide range of parameter settings, indicating the robustness of GDRT to hyperparameter selection.

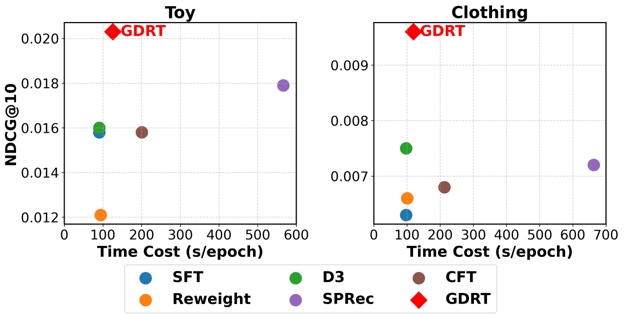

Figure 12: Performance comparisons in terms of both recommendation accuracy and efficiency.

4.5 Efficiency Comparison (RQ5)

This section compares the efficiency and performance of GDRT with baselines. As illustrated in Figure 12, GDRT demonstrates both optimal performance and high computational efficiency, incurring negligible overhead compared to SFT. In contrast, CFT introduces additional counterfactual samples leading to substantially higher computational costs. SPRec, based on DPO [48], requires an extra reference model and negative samples during training, resulting in a considerable increase in runtime. Although Reweight and D3 are relatively efficient, their performance improvements are limited.

5 Related Work

5.1 Sequential Recommendation

Sequential recommendation focuses on predicting the next item a user will be interested in based on their historical interactions. Compared to collaborative filtering [9, 10, 57, 69, 70, 74], sequential recommendation incorporates temporal information and places greater emphasis on capturing the evolving patterns of user interests. With the advancement of deep learning, numerous architectures based on deep neural networks have been introduced into the sequential recommendation. For example, GRU4Rec [26] employs RNNs, while Caser [53] utilizes CNNs to effectively capture long-term dependencies and modeling user interest patterns from historical behavior. More advanced models such as SASRec [30] and BERT4Rec [51] leverage self-attention mechanisms [56], enabling the identification of the most relevant parts within the sequence. Due to the dynamic nature of data distributions as time evolves [63, 64], DROS [71] introduces DRO [49, 67] to further enhance the model's robustness against distributional shifts caused by temporal changes. The readers may refer to the survey [20] for more details.

5.2 Biases in LLM-based Recommendation

Large Language Models (LLMs), with their powerful capabilities in comprehension, reasoning, and extensive knowledge [1, 19, 23, 55], have been widely applied to recommendation systems [14, 15, 59, 68]. One prominent paradigm is LLM-based RS [35], which directly leverages LLMs as the backbone of the recommender. Subsequent studies have explored fine-tuning LLMs on domain-specific recommendation datasets to further enhance their recommendation capabilities [2, 4, 12, 32, 35, 38, 58, 60, 81].

Recent studies have extensively explored bias and fairness issues in LLM-based RS, such as popularity bias [21, 22, 29, 37, 40, 41], position bias [6, 7, 17, 27, 28, 42, 43, 72], amplification bias [3], and bias stemming from LLMs' preferences for specific item attributes [18, 29, 36, 73]. Nevertheless, existing research has largely overlooked context bias, which arises during fine-tuning and reflects the inherent over-reliance of LLMs on auxiliary tokens. Since existing debiasing methods fail to account for this factor, their effectiveness remains limited. CFT [75] seeks to enhance the modeling of users' historical interactions, but suffers from objective misalignment, weight selection challenges, and high computational cost, restricting its applicability. These limitations are further discussed in Section 2.2.4.

5.3 Group Distributionally Robust Optimization

Group Distributionally Robust Optimization (Group DRO) [50] is an optimization framework that operates over predefined sample groups, aiming to achieve consistent and reliable performance across them by emphasizing the optimizing of the worst-performing group during training. It has been widely applied in various domains [46, 50, 77] and has demonstrated strong effectiveness in mitigating group disparities [25, 77] as well as reducing models' reliance on shortcut correlations [13, 24]. Several studies have applied Group DRO to RS. For example, S-DRO [65] uses group DRO to improve the experience of underrepresented user groups that tend to engage with less popular items. PDRO [77] extends this approach with popularity-aware mechanisms to prevent harming the performance of popular items.

6 Conclusion

In this work, we identify a key limitation of supervised fine-tuning (SFT) in LLM-based recommenders: it often induces **Context Bias**, whereby the model over-relies on auxiliary tokens (*e.g.,* task descriptions and prefix-generated tokens) while underutilizing core user interaction information. This bias undermines recommendation accuracy and raises unfairness concerns. To address this issue, we introduce Group Distributionally Robust Optimization-based Tuning (GDRT), which aims to reduce the model's over-reliance on auxiliary tokens by applying Group DRO across token groups with varying degrees of relevance to auxiliary tokens. Extensive experiments on multiple public datasets demonstrate that GDRT effectively mitigates context bias, thereby significantly improving recommendation accuracy and enhancing fairness.

This work investigates a novel form of bias introduced by the integration of LLMs into recommenders, which is not present in traditional recommendation models. A promising avenue for future work is to examine whether LLM-based RS exhibit additional, as-yet unidentified biases.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (62372399, 62476244), OPPO Research Fund, the Starry Night Science Fund of Zhejiang University Shanghai Institute for Advanced Study (SN-ZJU-SIAS-001), and the advanced computing resources provided by the Supercomputing Center of Hangzhou City University. We thank the reviewers for their valuable and insightful suggestions that improve the paper.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

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

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

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

[5] Oren Barkan, Yonatan Toib, Yehonatan Elisha, Jonathan Weill, and Noam Koenigstein. 2024. LLM Explainability via Attributive Masking Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. 9522–9537.

[6] Ethan Bito, Yongli Ren, and Estrid He. 2025. Evaluating Position Bias in Large Language Model Recommendations. *arXiv preprint arXiv:2508.02020* (2025).

[7] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems* 41, 3 (2023), 1–39.

[8] Lu Chen, Yizhou Wang, Shixiang Tang, Qianhong Ma, Tong He, Wanli Ouyang, Xiaowei Zhou, Hujun Bao, and Sida Peng. 2025. EgoAgent: A Joint Predictive Agent Model in Egocentric Worlds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 6970–6980.

[9] Sirui Chen, Jiawei Chen, Sheng Zhou, Bohao Wang, Shen Han, Chanfei Su, Yuqing Yuan, and Can Wang. 2024. SIGformer: Sign-aware Graph Transformer for Recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1274–1284.

[10] Sirui Chen, Shen Han, Jiawei Chen, Binbin Hu, Sheng Zhou, Gang Wang, Yan Feng, Chun Chen, and Can Wang. 2025. Rankformer: A Graph Transformer for Recommendation based on Ranking Objective. In *Proceedings of the ACM on Web Conference 2025*. 3037–3048.

[11] Sirui Chen, Changxin Tian, Binbin Hu, Kunlong Chen, Ziqi Liu, Zhiqiang Zhang, and Jun Zhou. 2025. Arrows of math reasoning data synthesis for large language models: Diversity, complexity and correctness. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*. 4665–4669.

[12] Yuxin Chen, Junfei Tan, An Zhang, Zhengyi Yang, Leheng Sheng, Enzhi Zhang, Xiang Wang, and Tat-Seng Chua. 2024. On Softmax Direct Preference Optimization for Recommendation. *arXiv preprint arXiv:2406.09215* (2024).

[13] Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. 2021. Environment inference for invariant learning. In *International Conference on Machine Learning*. PMLR, 2189–2200.

[14] Yu Cui, Feng Liu, Jiawei Chen, Canghong Jin, Xingyu Lou, Changwang Zhang, Jun Wang, Yuegang Sun, and Can Wang. 2025. HatLLM: Hierarchical Attention Masking for Enhanced Collaborative Modeling in LLM-based Recommendation. *arXiv preprint arXiv:2510.10955* (2025).

[15] Yu Cui, Feng Liu, Jiawei Chen, Xingyu Lou, Changwang Zhang, Jun Wang, Yuegang Sun, Xiaohu Yang, and Can Wang. 2025. Field Matters: A lightweight LLM-enhanced Method for CTR Prediction. *arXiv preprint arXiv:2505.14057* (2025).

[16] Yu Cui, Feng Liu, Pengbo Wang, Bohao Wang, Heng Tang, Yi Wan, Jun Wang, and Jiawei Chen. 2024. Distillation Matters: Empowering Sequential Recommenders to Match the Performance of Large Language Models. In *Proceedings of the 18th ACM Conference on Recommender Systems*. 507–517.

[17] Sunhao Dai, Chen Xu, Shicheng Xu, Liang Pang, Zhenhua Dong, and Jun Xu. 2024. Bias and unfairness in information retrieval systems: New challenges in the llm era. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 6437–6447.

[18] Yashar Deldjoo. 2024. Understanding biases in ChatGPT-based recommender systems: Provider fairness, temporal stability, and recency. *ACM Transactions on Recommender Systems* (2024).

[19] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The Llama 3 Herd of Models. *arXiv preprint arXiv:2407.21783* (2024).

[20] Hui Fang, Danning Zhang, Yiheng Shu, and Guibing Guo. 2020. Deep learning for sequential recommendation: Algorithms, influential factors, and evaluations. *ACM Transactions on Information Systems (TOIS)* 39, 1 (2020), 1–42.

[21] Chongming Gao, Ruijun Chen, Shuai Yuan, Kexin Huang, Yuanqing Yu, and Xiangnan He. 2025. Sprec: Self-play to debias llm-based recommendation. In *Proceedings of the ACM on Web Conference 2025*. 5075–5084.

[22] Chongming Gao, Mengyao Gao, Chenxiao Fan, Shuai Yuan, Wentao Shi, and Xiangnan He. 2025. Process-supervised llm recommenders via flow-guided tuning. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1934–1943.

[23] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. 2025. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature* 645, 8081 (2025), 633–638.

[24] Hyeonggeun Han, Sehwan Kim, Hyungjun Joo, Sangwoo Hong, and Jungwoo Lee. 2024. Mitigating spurious correlations via disagreement probability. *Advances in Neural Information Processing Systems* 37 (2024), 74363–74382.

[25] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. 2018. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*. PMLR, 1929–1938.

[26] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939* (2015).

[27] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2024. Large language models are zero-shot rankers for recommender systems. In *European Conference on Information Retrieval*. Springer, 364–381.

[28] Chumeng Jiang, Jiayin Wang, Weizhi Ma, Charles LA Clarke, Shuai Wang, Chuhan Wu, and Min Zhang. 2025. Beyond Utility: Evaluating LLM as Recommender. In *Proceedings of the ACM on Web Conference 2025*. 3850–3862.

[29] Meng Jiang, Keqin Bao, Jizhi Zhang, Wenjie Wang, Zhengyi Yang, Fuli Feng, and Xiangnan He. 2024. Item-side fairness of large language model-based recommendation system. In *Proceedings of the ACM Web Conference 2024*. 4717–4726.

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

[31] Carina Kauf, Emmanuele Chersoni, Alessandro Lenci, Evelina Fedorenko, and Anna Ivanova. 2024. Log Probabilities Are a Reliable Estimate of Semantic Plausibility in Base and Instruction-Tuned Language Models. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*. 263–277.

[32] Sein Kim, Hongseok Kang, Seungyoon Choi, Donghyun Kim, Minchul Yang, and Chanyoung Park. 2024. Large language models meet collaborative filtering: An efficient all-round llm-based recommender system. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1395–1406.

[33] Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, et al. 2020. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896* (2020).

[34] Simon Lepage, Jeremie Mary, and David Picard. 2025. Closing the Performance Gap in Generative Recommenders with Collaborative Tokenization and Efficient Modeling. *arXiv preprint arXiv:2508.14910* (2025).

[35] Lei Li, Yongfeng Zhang, Dugang Liu, and Li Chen. 2023. Large language models for generative recommendation: A survey and visionary discussions. *arXiv preprint arXiv:2309.01157* (2023).

[36] Xinyi Li, Yongfeng Zhang, and Edward C Malthouse. 2023. A preliminary study of chatgpt on news recommendation: Personalization, provider fairness, fake news. *arXiv preprint arXiv:2306.10702* (2023).

[37] Jiayi Liao, Xiangnan He, Ruobing Xie, Jiancan Wu, Yancheng Yuan, Xingwu Sun, Zhanhui Kang, and Xiang Wang. 2024. RosePO: Aligning LLM-based Recommenders with Human Values. *arXiv preprint arXiv:2410.12519* (2024).

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

[39] Jianghao Lin, Rong Shan, Chenxu Zhu, Kounianhua Du, Bo Chen, Shigang Quan, Ruiming Tang, Yong Yu, and Weinan Zhang. 2024. Rella: Retrieval-enhanced large language models for lifelong sequential behavior comprehension in recommendation. In *Proceedings of the ACM on Web Conference 2024*. 3497–3508.

[40] Siyi Lin, Chongming Gao, Jiawei Chen, Sheng Zhou, Binbin Hu, Yan Feng, Chun Chen, and Can Wang. 2025. How do recommendation models amplify popularity bias? An analysis from the spectral perspective. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*. 659–668.

[41] Sijin Lu, Zhibo Man, Fangyuan Luo, and Jun Wu. 2025. Dual Debiasing in LLM-based Recommendation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2685–2689.

[42] Sichun Luo, Bowei He, Haohan Zhao, Wei Shao, Yanlin Qi, Yinya Huang, Aojun Zhou, Yuxuan Yao, Zongpeng Li, Yuanzhang Xiao, et al. 2025. Recranker: Instruction tuning large language model as ranker for top-k recommendation. *ACM Transactions on Information Systems* 43, 5 (2025), 1–31.

[43] Tianhui Ma, Yuan Cheng, Hengshu Zhu, and Hui Xiong. 2023. Large language models are not stable recommender systems. *arXiv preprint arXiv:2312.15746* (2023).

[44] Vivek Miglani, Aobo Yang, Aram H Markosyan, Diego Garcia-Olano, and Narine Kokhlikyan. 2023. Using captum to explain generative language models. *arXiv preprint arXiv:2312.05491* (2023).

[45] Hyunsoo Na, Minseok Gang, Youngrok Ko, Jinseok Seol, and Sang-goo Lee. 2024. Enhancing Large Language Model Based Sequential Recommender Systems with Pseudo Labels Reconstruction. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. 7213–7222.

[46] Yonatan Oren, Shiori Sagawa, Tatsunori B Hashimoto, and Percy Liang. 2019. Distributionally Robust Language Modeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 4227–4237.

[47] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems* 35 (2022), 27730–27744.

[48] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2024. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36 (2024).

[49] Hamed Rahimian and Sanjay Mehrotra. 2019. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659* (2019).

[50] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. 2019. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731* (2019).

[51] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1441–1450.

[52] Juntao Tan, Shuyuan Xu, Wenyue Hua, Yingqiang Ge, Zelong Li, and Yongfeng Zhang. 2024. Idgenrec: Llm-recsys alignment with textual id learning. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*. 355–364.

[53] Jiaxi Tang and Ke Wang. 2018. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the eleventh ACM international conference on web search and data mining*. 565–573.

[54] Jiabin Tang, Yuhao Yang, Wei Wei, Lei Shi, Lixin Su, Suqi Cheng, Dawei Yin, and Chao Huang. 2024. Graphgpt: Graph instruction tuning for large language models. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 491–500.

[55] Qwen Team et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671* 2, 3 (2024).

[56] Ashish Vaswani. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762* (2017).

[57] Bohao Wang, Jiawei Chen, Changdong Li, Sheng Zhou, Qihao Shi, Yang Gao, Yan Feng, Chun Chen, and Can Wang. 2024. Distributionally Robust Graph-based Recommendation System. In *Proceedings of the ACM on Web Conference 2024*. 3777–3788.

[58] Bohao Wang, Feng Liu, Jiawei Chen, Xingyu Lou, Changwang Zhang, Jun Wang, Yuegang Sun, Yan Feng, Chun Chen, and Can Wang. 2025. Msl: Not all tokens are what you need for tuning llm as a recommender. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1912–1922.

[59] Bohao Wang, Feng Liu, Changwang Zhang, Jiawei Chen, Yudi Wu, Sheng Zhou, Xingyu Lou, Jun Wang, Yan Feng, Chun Chen, et al. 2025. Llm4dsr: Leveraging large language model for denoising sequential recommendation. *ACM Transactions on Information Systems* 44, 1 (2025), 1–32.

[60] Hangyu Wang, Jianghao Lin, Xiangyang Li, Bo Chen, Chenxu Zhu, Ruiming Tang, Weinan Zhang, and Yong Yu. 2024. Flip: Fine-grained alignment between id-based models and pretrained language models for ctr prediction. In *Proceedings of the 18th ACM conference on recommender systems*. 94–104.

[61] Lei Wang and Ee-Peng Lim. 2023. Zero-shot next-item recommendation using large pretrained language models. *arXiv preprint arXiv:2304.03153* (2023).

[62] Yu Wang, Junshu Dai, Yuchen Ying, Yuxuan Liang, Tongya Zheng, and Mingli Song. 2025. Adaptive Location Hierarchy Learning for Long-Tailed Mobility Prediction. *arXiv preprint arXiv:2505.19965* (2025).

[63] Yu Wang, Tongya Zheng, Yuxuan Liang, Shunyu Liu, and Mingli Song. 2024. Cola: Cross-city mobility transformer for human trajectory simulation. In *Proceedings of the ACM on Web Conference 2024*. 3509–3520.

[64] Yu Wang, Tongya Zheng, Shunyu Liu, Zunlei Feng, Kaixuan Chen, Yunzhi Hao, and Mingli Song. 2024. Spatiotemporal-Augmented Graph Neural Networks for Human Mobility Simulation. *IEEE Transactions on Knowledge and Data Engineering* 36, 11 (2024), 7074–7086.

[65] Hongyi Wen, Xinyang Yi, Tiansheng Yao, Jiaxi Tang, Lichan Hong, and Ed H Chi. 2022. Distributionally-robust recommendations for improving worst-case user experience. In *Proceedings of the ACM Web Conference 2022*. 3606–3610.

[66] Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. 2023. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671* (2023).

[67] Junkang Wu, Jiawei Chen, Jiancan Wu, Wentao Shi, Xiang Wang, and Xiangnan He. 2023. Understanding contrastive learning via distributionally robust optimization. *Advances in Neural Information Processing Systems* 36 (2023), 23297–23320.

[68] Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, et al. 2024. A survey on large language models for recommendation. *World Wide Web* 27, 5 (2024), 60.

[69] Weiqin Yang, Jiawei Chen, Xin Xin, Sheng Zhou, Binbin Hu, Yan Feng, Chun Chen, and Can Wang. 2024. PSL: Rethinking and Improving Softmax Loss from Pairwise Perspective for Recommendation. *arXiv preprint arXiv:2411.00163* (2024).

[70] Weiqin Yang, Jiawei Chen, Shengjia Zhang, Peng Wu, Yuegang Sun, Yan Feng, Chun Chen, and Can Wang. 2025. Breaking the top-k barrier: Advancing top-k ranking metrics optimization in recommender systems. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*. 3542–3552.

[71] Zhengyi Yang, Xiangnan He, Jizhi Zhang, Jiancan Wu, Xin Xin, Jiawei Chen, and Xiang Wang. 2023. A generic learning framework for sequential recommendation with distribution shifts. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 331–340.

[72] Junjie Zhang, Yupeng Hou, Ruobing Xie, Wenqi Sun, Julian McAuley, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2024. Agentcf: Collaborative learning with autonomous language agents for recommender systems. In *Proceedings of the ACM Web Conference 2024*. 3679–3689.

[73] Jiaming Zhang, Yuyuan Li, Yiqun Xu, Li Zhang, Xiaohua Feng, Zhifei Ren, and Chaochao Chen. 2025. BiFair: A Fairness-aware Training Framework for LLM-enhanced Recommender Systems via Bi-level Optimization. *arXiv preprint arXiv:2507.04294* (2025).

[74] Shengjia Zhang, Jiawei Chen, Changdong Li, Sheng Zhou, Qihao Shi, Yan Feng, Chun Chen, and Can Wang. 2025. Advancing Loss Functions in Recommender Systems: A Comparative Study with a Rényi Divergence-Based Solution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 13286–13294.

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

[76] Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology* 15, 2 (2024), 1–38.

[77] Jujia Zhao, Wenjie Wang, Xinyu Lin, Leigang Qu, Jizhi Zhang, and Tat-Seng Chua. 2023. Popularity-aware distributionally robust optimization for recommendation system. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 4967–4973.

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

[79] Zhi Zheng, Wenshuo Chao, Zhaopeng Qiu, Hengshu Zhu, and Hui Xiong. 2024. Harnessing large language models for text-rich sequential recommendation. In *Proceedings of the ACM Web Conference 2024*. 3207–3216.

[80] Wei Zhou, Heike Adel, Hendrik Schuff, and Ngoc Thang Vu. 2024. Explaining pre-trained language models with attribution scores: An analysis in low-resource settings. *arXiv preprint arXiv:2403.05338* (2024).

[81] Yaochen Zhu, Liang Wu, Qi Guo, Liangjie Hong, and Jundong Li. 2024. Collaborative large language model for recommender systems. In *Proceedings of the ACM on Web Conference 2024*. 3162–3172.

A Appendices

A.1 Additional Validation of Context Bias Across Prompt Templates and LLMs

In this section, to further rule out the potential influence of prompt templates and different LLM backbones on the analysis of context bias, we extend the FAA experiments described in Section 2.2.1 to a broader range of prompt templates (see Figure 13) and LLM backbones (see Figure 14), beyond those used in the main experiments. Specifically, for prompt templates, we adopt those proposed in [38] and [3], which we refer to as *Prompt1* and *Prompt2*, respectively. For LLM backbones, we evaluate LLaMA3-8B [19] and Qwen2.5-1.5B [55]. Across all prompt template and backbone configurations, SFT consistently amplifies the ratio of attribution values between auxiliary tokens and user-interaction tokens. This observation indicates that context bias persistently exists across different prompt templates and LLM backbones. The underlying reason is that the origin of context bias lies in the dataset itself: the co-occurrence rate between auxiliary tokens and target item tokens is significantly higher than that between interaction tokens and target item tokens, as discussed in Section 2.2.3.

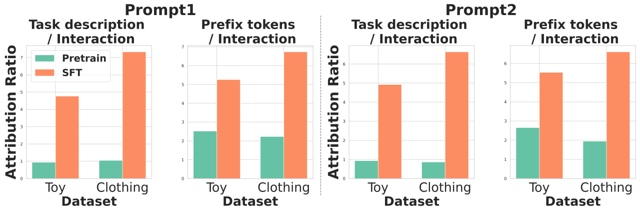

Figure 13: Ratio of attribution values between auxiliary tokens and user-interaction tokens before and after SFT across different prompt templates.

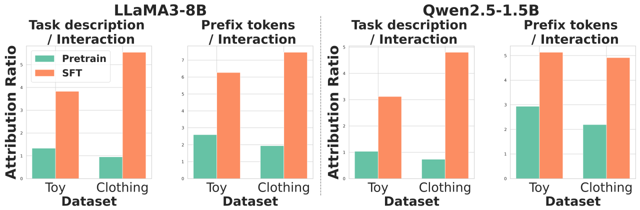

Figure 14: Ratio of attribution values between auxiliary tokens and user-interaction tokens before and after SFT across different LLMs.

A.2 Details of Baselines

The methods compared fall into several categories:

- **Traditional RS (SASRec [30], SASRec++ [34], DROS [71])**: SASRec is a representative sequential recommendation model that employs self-attention mechanisms to effectively capture users' dynamic interest patterns from historical interaction sequences. Building upon SASRec, SASRec++ introduces an improved training objective by adopting the softmax loss instead of BCE loss used in original SASRec, which leads to more stable optimization and enhanced recommendation performance. DROS incorporates Distributionally Robust Optimization (DRO) into sequential recommendation, aiming to improve model robustness under distributional shifts.
- **LLM-based RS (SFT [2], CFT [75], MSL [58], LLaRA [38], A-LLM [32])**: This line of work leverages the strong representation and reasoning capabilities of LLMs for recommendation. SFT applies instruction-tuning strategies with carefully designed templates to adapt LLMs to recommendation tasks. CFT incorporates a causal loss to strengthen the behavior sequence modeling capabilities of LLMs. MSL improves the loss function specifically tailoring it to optimize recommendation-oriented objectives. LLaRA enhances LLM-based recommenders by incorporating embeddings from traditional recommendation models, enabling better exploitation of collaborative filtering signals. A-LLM extends this idea by aligning these collaborative embeddings with their corresponding textual semantics, facilitating more effective integration of structured and unstructured information.
- **Debiasing Methods for LLM-based RS (Reweight [29], SPRec [21], D3 [3])**: These methods focus on mitigating various biases that arise when applying LLMs to recommendation. Reweight addresses popularity bias by balancing recommendations using pre-calculated item weights. SPRec proposes a popularity-aware negative sampling strategy within Direct Preference Optimization (DPO) [48] to reduce popularity bias. D3 focuses on mitigating amplification bias during inference by improving the decoding strategy, preventing the model from over-recommending items whose textual representations contain tokens with excessively high generation probabilities.

A.3 Additional Performance Comparison Across Prompt Templates and LLMs

In this section, we present additional comparative experiments on the performance of GDRT and SFT that go beyond the prompt templates and LLM backbones used in the main experiments. As summarized in Tables 4 and 5, GDRT consistently improves recommendation accuracy while achieving substantial gains in fairness across all evaluated configurations, demonstrating strong generalization ability over a broader range of prompts and LLMs.

Table 4: Performance comparison of SFT and GDRT across different prompt templates. The best result is bolded.

Prompt	Dataset	Method	NDCG@5	DGU@5
Prompt1 [38]	Toy	SFT	0.0118	0.6549
		GDRT	**0.0144**	**0.4616**
	Clothing	SFT	0.0033	0.4631
		GDRT	**0.0052**	**0.2025**
Prompt2 [3]	Toy	SFT	0.0116	0.5765
		GDRT	**0.0136**	**0.3987**
	Clothing	SFT	0.0039	0.5231
		GDRT	**0.0045**	**0.1893**

Table 5: Performance comparison of SFT and GDRT across different LLMs. The best result is bolded.

LLM	Dataset	Method	NDCG@5	DGU@5
Llama3-8B	Toy	SFT	0.0151	0.6861
		GDRT	**0.0173**	**0.5970**
	Clothing	SFT	0.0039	0.5801
		GDRT	**0.0062**	**0.2068**
Qwen2.5-1.5B	Toy	SFT	0.0098	0.4849
		GDRT	**0.0117**	**0.2311**
	Clothing	SFT	0.0018	0.3538
		GDRT	**0.0026**	**0.0778**

A.4 The proof of Lemma 1

The original formulation of GDRT

$$\mathcal{L}_{GDRT} = \max_{Q} \sum_{g=1}^{G} Q(g)\mathcal{L}(g), \quad \text{s.t. } D_{KL}(Q, U) \leq \eta, \tag{6}$$

can be rewritten in the expectation form as

$$\begin{aligned} &\max_{Q} \mathbb{E}_{g\sim Q}[\mathcal{L}(g)] \\ &s.t. \mathbb{E}_{g\sim Q}[\log \frac{Q(g)}{U(g)}] \leq \eta \end{aligned} \tag{7}$$

In the following, we focus on how to eliminate the inner maximization optimization problem and the KL constraint term. Assume $W(g) = Q(g)/U(g)$ and define a convex function $\phi(x) = x\log x - x + 1$. Then the divergence $D_{KL}(Q, U)$ can be written as $\mathbb{E}_U[\phi(W)]$. The inner layer maximization optimization problem can be reformulated as follow:

$$\begin{aligned} &\max_{W} \mathbb{E}_U\,[\mathcal{L}W] \\ &\text{s.t. } \mathbb{E}_U[\phi(W)] \leq \eta, \mathbb{E}_U[W] = 1 \end{aligned} \tag{8}$$

As a convex optimization problem, we use the Lagrangian function to solve it:

$$\begin{aligned} &\min_{\tau\geq 0,\beta} \max_{W} \mathbb{E}_U\,[\mathcal{L}W] - \tau(\mathbb{E}_U[\phi(W)] - \eta) + \beta(\mathbb{E}_U[W] - 1) \\ &= \min_{\tau\geq 0,\beta} \left\{\tau\eta - \beta + \tau \max_{W} \mathbb{E}_U \left[\frac{\mathcal{L}+\beta}{\tau}W - \phi(W)\right]\right\} \\ &= \min_{\tau\geq 0,\beta} \left\{\tau\eta - \beta + \tau\mathbb{E}_U \left[\max_{W}\left(\frac{\mathcal{L}+\beta}{\tau}W - \phi(W)\right)\right]\right\} \end{aligned} \tag{9}$$

Notice that $\max_W \left(\frac{\mathcal{L}+\beta}{\tau}W - \phi(W)\right) = \phi^*(\frac{\mathcal{L}+\beta}{\tau})$ is the convex conjugate function of $\phi(x)$ and we have $\phi^*(x) = e^x - 1$. $W(g) = e^{\frac{\mathcal{L}(g)+\beta}{\tau}}$ when the maximum value is obtained.

$$\begin{aligned} &\min_{\tau\geq 0,\beta} \left\{\tau\eta - \beta + \tau\mathbb{E}_U \left[\max_{W}\left(\frac{\mathcal{L}+\beta}{\tau}W - \phi(W)\right)\right]\right\} \\ &= \min_{\tau\geq 0,\beta} \left\{\tau\eta - \beta + \tau\mathbb{E}_U \left[e^{\frac{\mathcal{L}+\beta}{\tau}} - 1\right]\right\} \\ &= \min_{\tau\geq 0} \left\{\tau\eta + \tau\log\mathbb{E}_U \left[e^{\frac{\mathcal{L}}{\tau}}\right]\right\} \end{aligned} \tag{10}$$

where $\beta = -\tau\log\mathbb{E}_{g\sim U}\left[e^{\frac{\mathcal{L}(g)}{\tau}}\right]$ and $W(g) = \frac{e^{\frac{\mathcal{L}(g)}{\tau}}}{\mathbb{E}_{g'\sim U}\left[e^{\frac{\mathcal{L}(g')}{\tau}}\right]}$ when the minimum value is obtained. We consider the Lagrange multiplier τ as a hyperparameter related to the robustness radius η. Then we can get the unconstrained optimization problem as follows,

$$\mathcal{L}_{GDRT} = \tau\eta + \tau\log\mathbb{E}_{g\sim U}\exp\left(\frac{\mathcal{L}(g)}{\tau}\right) \tag{11}$$

where the worst-case distribution

$$Q^*(g) = U(g)\frac{\exp\left(\mathcal{L}(g)/\tau\right)}{\mathbb{E}_{i\sim U}\left[\exp(\mathcal{L}(i)/\tau)\right]} \tag{12}$$

Since U is a uniform distribution

$$Q^*(g) = \frac{\exp\left(\mathcal{L}(g)/\tau\right)}{\sum_{g'}\left[\exp(\mathcal{L}(g')/\tau)\right]} \tag{13}$$

Thus lemma 1 is proven.