Two-sided Calibration for Quality-aware Responsible Recommendation

CHENYANG WANG, DCST, BNRist, Tsinghua University, China
YANKAI LIU, China Mobile Research Institute & THU-CMCC Joint Institute, China
YUANQING YU, DCST, BNRist, Tsinghua University, China
WEIZHI MA*, AIR, Tsinghua University, China
MIN ZHANG*, DCST, BNRist, Tsinghua University & THU-CMCC Joint Institute, China
YIQUN LIU, DCST, BNRist, Tsinghua University, China
HAITAO ZENG, China Mobile Research Institute, China
JUNLAN FENG, China Mobile Research Institute, China
CHAO DENG, China Mobile Research Institute, China

Calibration in recommender systems ensures that the user's interests distribution over groups of items is reflected with their corresponding proportions in the recommendation, which has gained increasing attention recently. For example, a user who watched 80 entertainment videos and 20 knowledge videos is expected to receive recommendations comprising about 80% entertainment and 20% knowledge videos as well. However, with the increasing calls for responsible recommendation, it has become inadequate to just match users' historical behaviors especially when items are grouped by their qualities, which could result in undesired effects at the system level (e.g., overwhelming clickbaits). In this paper, we envision the *two-sided calibration* task that not only matches the users' past interests distribution (user-level calibration) but also guarantees an overall target exposure distribution of different item groups (system-level calibration). The target group exposure distribution can be explicitly pursued by users, platform owners, and even the law (e.g., the platform owners expect about 50% knowledge video recommendation on the whole). To support this scenario, we propose a post-processing method named PCT. PCT first solves personalized calibration targets that minimize the changes in users' historical interest distributions while ensuring the overall target group exposure distribution. Then, PCT reranks the original recommendation lists according to personalized calibration targets to generate both relevant and two-sided calibrated recommendations. Extensive experiments demonstrate the superior performance of the proposed method compared to calibrated and fairness-aware recommendation approaches.

CCS Concepts: • **Information systems** → **Recommender systems**.

Additional Key Words and Phrases: recommender systems, two-sided calibration, quality-aware responsible recommendation

ACM Reference Format:
Chenyang Wang, Yankai Liu, Yuanqing Yu, Weizhi Ma, Min Zhang, Yiqun Liu, Haitao Zeng, Junlan Feng, Chao Deng. 2023. Two-sided Calibration for Quality-aware Responsible Recommendation. In *Seventeenth ACM Conference on Recommender Systems (RecSys '23), September 18–22, 2023, Singapore, Singapore.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3604915.3608799

*Corresponding authors.
This work is supported by the Natural Science Foundation of China (Grant No. U21B2026, 62002191) and Tsinghua University Guoqiang Research Institute.

Manuscript submitted to ACM

1 INTRODUCTION

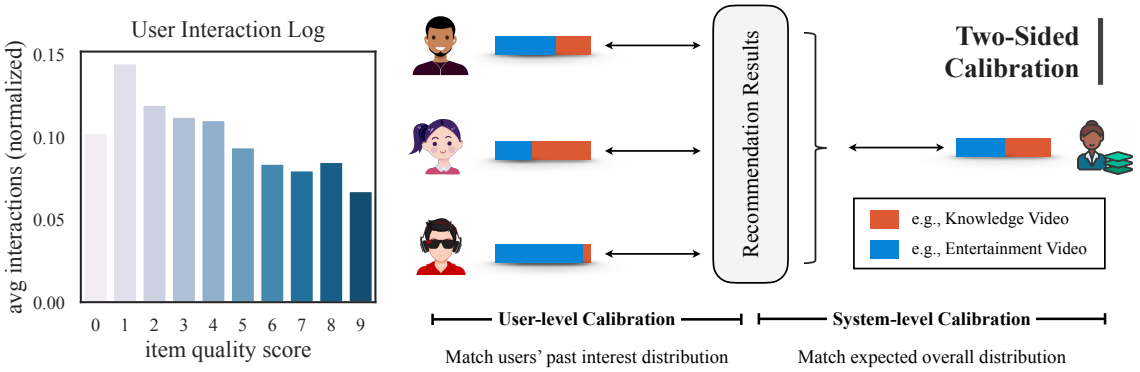
Recommender systems play an important role in supporting various human decision-making activities in e-commerce, entertainment, and social networks. Early recommendation methods usually focus on maximizing accuracy and enhancing personalization [10, 32]. Although such a paradigm leads to recommendation results with high interaction rates, recommender systems could bring undesirable effects (e.g., misinformation spreading [9], group polarization [7]) without appropriate technical interventions. Thus, there has been increasing attention paid to beyond-accuracy characteristics of recommendations, such as diversity [30], novelty [6], fairness [17], and so on [3, 8].

One of the critical characteristics of recommender systems is *calibration*, which has gained rising attention recently [1, 21, 27, 29]. Calibrated recommendation aims to generate a recommendation list that consists of items in the same proportion of the topics the user has previously liked [27]. For example, if a user's interaction history contains 80 entertainment videos and 20 knowledge videos, calibrated recommendation expects the top-k recommendation list comprises 80% entertainment and 20% knowledge videos as well. Calibration ensures that the various (past) areas of interest of a user are reflected with their corresponding proportions [29]. This is of great importance to avoid the dominance of users' main interests and maintain consistency with users' historical interest distributions.

However, with the increasing calls for responsible recommendation [3, 8], it has become inadequate to only calibrate the recommendation results according to users' historical interactions, which might still result in undesired platform ecology from the system perspective. For example, low-quality items (e.g., clickbaits, misinformation, etc.) usually have higher clicking persuasion but lead to negative impacts on users' experiences [12, 18]. If most users are dedicated to low-quality items, high-quality items can hardly get proper exposure due to the calibration towards users' interest distributions. We validate this on a recently published dataset Tenrec [38], where each item (i.e., article) is manually annotated with a quality score (from 0 to 9, higher scores are better) based on its content. In this news streaming scenario, Figure 1(a) shows that users indeed interact with more low-quality items on average. As a result, calibrated recommendation could still lead to unhealthy platform ecology dominated by low-quality items.

To this end, it is important to additionally regulate the exposure proportion of high-quality items at the system level [12, 18, 34]. In this paper, we envision the *two-sided calibration* task that not only matches the users' past interest distribution (user-level calibration) but also guarantees an overall target group exposure distribution (system-level calibration), as shown in Figure 1(b). The target exposure distribution could be explicitly pursued by users, platform owners, and even the law. For example, an online learning platform may expect a certain ratio of exposure to authoritative courses. Compared to previous calibrated recommendation studies [1, 27, 29], the two-sided calibration task additionally seeks to ensure a given overall target group exposure distribution, which enables systemic regulation of different groups of items. Compared to some fairness-aware studies [19, 35, 36], although they can also achieve systemic group exposure regulation, they do not take user-level calibration into consideration. Previous work [29] has shown that user-level calibration is a complementary notion of fairness and is also important for responsible recommendation.

In practice, the two-sided calibration task faces two main challenges: 1) how to determine the target group exposure distribution for each user to optimize user-level calibration while ensuring system-level calibration? 2) how to generate the final recommendation result for each user that is both relevant to his/her interests and close to the personalized target group exposure distribution? First, different from previous calibrated recommendation studies that directly treat the user's historical interest distribution as the target exposure distribution, it is non-trivial for two-sided calibration to derive the target exposure distribution for each user under the restriction of system-level calibration. Second, regulating



(a) Users' Interest Distribution (b) Two-sided Calibration Task

Fig. 1. (a) Average interactions per item for items with different quality scores in the Tenrec dataset. Users interact with more low-quality items, in which case calibrated recommendation still leads to unhealthy platform ecology. (b) Illustration of the proposed two-sided calibration task, which not only matches users' past interest distributions (user-level calibration) but also ensures an expected overall distribution (system-level calibration).

the group exposure distribution for each user is likely to hurt the ranking performance, which needs careful algorithm designs to achieve a better tradeoff.

To support the two-sided calibration task, we propose a post-processing method based on Personalized Calibration Targets (PCT), which consists of two modules: PCT-Solver and PCT-Reranker. First, PCT-Solver focuses to solve personalized target group exposure distributions that minimize the changes in users' historical interest distributions while ensuring the overall target group exposure distribution, which can be formulated as a linear programming problem. Then, PCT-Reranker reranks the original recommendation lists according to these personalized calibration targets to generate both relevant and two-sided calibrated recommendations. Extensive experiments show that PCT can better balance various objectives of recommendation (ranking performance, item coverage, calibration) compared to existing calibrated and fairness-aware methods. The main contributions of this work can be summarized as follows:

- To the best of our knowledge, we are the first to investigate the two-sided calibration task, which additionally ensures a given overall target exposure distribution besides typical user-level calibration.
- We propose a post-processing method based on personalized calibration targets (PCT) to better support the two-sided calibration task.
- Extensive experiments on public and industrial datasets show that PCT can better balance different objectives of recommendation while achieving system-level calibration.

2 RELATED WORK

2.1 Calibrated Recommendation

Calibrated recommendation is first introduced in [22] and made widely aware by [29]. In brief, calibration in recommender systems measures whether the recommendations delivered to a user are consistent with the distribution of items the user has previously consumed. For example, if a user has watched 80% entertainment videos and 20% knowledge videos, the user might expect to see a similar distribution in the recommendation. Typical recommendation methods that focus on accuracy can easily lead to the dominance of users' main interests, which makes calibrated

recommendation of great importance in practice. To achieve calibrated recommendation, existing methods are mainly based on reranking. Steck [29] uses a greedy optimization that starts with an empty recommendation set and iteratively adds items to it if adding that item makes the list more relevant and calibrated. Seymen et al. [27] formulated calibrated recommendation as a mixed integer program with an L1-norm-based miscalibration penalty. Abdollahpouri et al. [1] further solve the calibration problem based on minimum-cost flow.

However, all the previous calibrated recommendation methods only focus on user-level calibration, which can not guarantee the overall content quality of presented items (e.g., clickbait). With the increasing calls for quality-aware responsible recommendation, it has become inadequate to just match the user's historical interest distribution. Differently, the proposed two-sided calibration task additionally pursues system-level calibration beyond user-level calibration, which enables systemic exposure regulation of different groups of items.

2.2 Fairness-aware Recommendation

Fairness is another important beyond-accuracy characteristic in recommendation. The concrete definition of fairness varies across different studies. According to stakeholders considered in the algorithm, there are user-side fairness [15], item-side fairness [14, 20], and two-sided fairness [35, 36]. User-side fairness usually aims at eliminating discrimination suffered by some users and encourages similar ranking performances across users. Item-side fairness usually focuses on providing fair exposure opportunities for different items. The expected item exposure is generally determined by the item's interaction rates (e.g., CTR). Two-sided fairness combines the above two lines of work and pursues the balance between user-side fairness and item-side fairness. Furthermore, according to the granularity of fairness, there exist individual fairness [25] and group fairness [2].

The proposed two-sided calibration task is related to but differs from existing fairness-aware studies. Although item-side/two-sided group fairness also regulates overall group exposure (similar to system-level calibration), it does not consider user-level calibration. Note that user-side fairness (similar ranking performance across users) cannot lead to user-level calibration (close item distribution for each user). Previous work [29] has shown that user-level calibration is a complementary notion of fairness and is also important for responsible recommendation.

2.3 Quality-aware Recommendation

The *quality* of an item in this work is determined by its content but not user interactions. For example, clickbaits usually have high click through rates but are considered low-quality[1] because their contents are non-informative. There have been a few studies focusing on the topic of quality-aware recommendation. Existing works can be categorized into two folds: quality effects analyses and quality modeling. On the one hand, more and more platforms are inclined to annotate item qualities according to their contents and investigate the quality effects. In the mobile news streaming scenario, Lu et.al. [18] identify the effects of the quality of news on user preferences and user behaviors. Users are more likely to click on low-quality news because it has a more attractive title, but when reading low-quality news, users usually read less with fewer revisits. Further, Iizuka et.al. [12] investigate the effects of news article quality on ad consumption based on quality annotations according to authenticity, value, expression, and headline. Sessions with high-quality news exposure are shown to have more ad consumption than sessions with low-quality news. On the other hand, some studies aim to model item quality based on users' implicit feedback. Early attempts mainly focus to identify low-quality or even harmful items (e.g., clickbaits) and filter these items in the ranking phase [24]. Other works

[1]The concrete standard to define quality may differ from application scenarios, which is not the focus of this paper.

use behavior signals (e.g., dwell time) as proxies for quality modeling, which serves as another objective beyond the original relevance prediction [34, 37].

However, with these item quality annotations (either human labeling or behavior estimation), existing works only implicitly improve the exposure quality. It still lacks investigation on how to ensure a given degree of exposure to high-quality items, which can be encompassed by the proposed two-sided calibration task when items are grouped by their quality.

3 PROBLEM FORMULATION

In this section, we formalize how to measure user-level calibration and system-level calibration. The proposed two-sided calibration task aims to generate recommendation results that pursue both user-level and system-level calibration.

Let $\mathcal{U}$ and $\mathcal{I}$ represent the set of users and items, respectively. We consider an attribute (e.g., quality) that categorizes items into different groups, where the attribute value takes from a finite set $\mathcal{G}$ (e.g., $\mathcal{G} = \{low, medium, high\}$). The mapping function $H : \mathcal{I} \rightarrow \mathcal{G}$ returns the group of a given item.

3.1 User-level Calibration

For each user $u \in \mathcal{U}$, we denote the interacted item set as I_u and the top-K recommendation list as R_u. Then, we can define the historical interest distribution $\mathbf{p}_u$ and the group exposure distribution $\mathbf{q}_u$ as follows:

$$\mathbf{p}_u(g) = \frac{\sum_{i \in I_u} I\left(H(i) = g\right)}{|I_u|}, \quad \mathbf{q}_u(g) = \frac{\sum_{k=1}^{K} r_k \cdot I\left(H(R_u[k]) = g\right)}{\sum_{k=1}^{K} r_k}, \quad g \in \mathcal{G}. \tag{1}$$

Here $I(\cdot)$ is an indicator function that only returns 1 if the condition is true, and r_k is the weight at rank k. Each element $\mathbf{p}_u(g), \mathbf{q}_u(g) \in [0, 1]$ and $\sum_{g \in \mathcal{G}} \mathbf{p}_u(g) = 1$, $\sum_{g \in \mathcal{G}} \mathbf{q}_u(g) = 1$.

The historical interest distribution $\mathbf{p}_u(g)$ measures the ratio of item group g the user u has interacted. The group exposure distribution $\mathbf{q}_u$ measures the rank-weighted ratio of item group g in the top-K recommendation list R_u. The simplest r_k can be equal across rankings (i.e., $r_k = 1$). Considering that the rankings of recommended items usually affect the examination probability, we use a decreasing ranking weight by default (i.e., $r_k = 1/\log_2(k + 1)$).

Then, the user-level calibration can be defined as follows:

Definition 1 (User-level Calibration). *The recommendation is user-level calibrated if the group exposure distribution is close to the historical interest distribution for each user.*

$$\mathbf{q}_u = \mathbf{p}_u, \quad \forall u \in \mathcal{U}.$$

In practice, we can use various disparity metrics to measure the distance between $\mathbf{q}_u$ and $\mathbf{p}_u$, such as the Kullback-Leibler (KL) divergence and Hellinger distance.

3.2 System-level Calibration

We define the overall group exposure distribution $\mathbf{q}$ as the average across users:

$$\mathbf{q} = \frac{1}{|\mathcal{U}|} \sum_{u=1}^{|\mathcal{U}|} \mathbf{q}_u \tag{2}$$

Besides the user-level calibration, system-level calibration additionally pursues a given target distribution $\hat{\mathbf{q}}$. Here the overall target group exposure distribution $\hat{\mathbf{q}}$ can be determined by different stakeholders (e.g., platform owners, law).

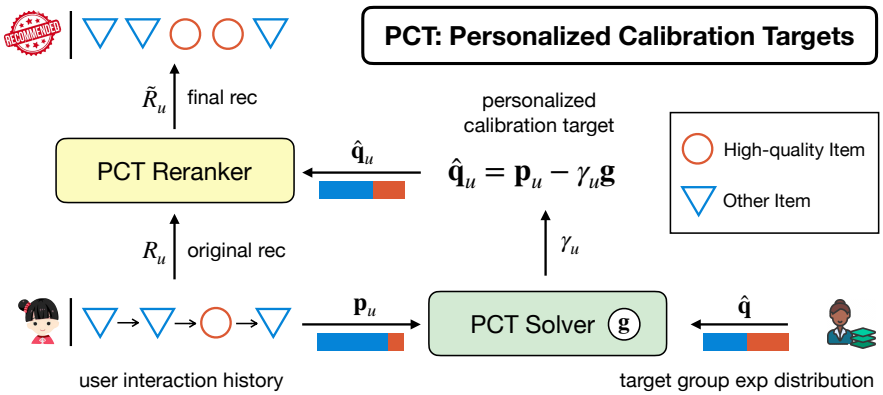


Fig. 2. Overview of the proposed PCT method. There are two main modules in PCT: 1) the PCT-Solver solves personalized target exposure distributions $\hat{\mathbf{q}}_u$ that minimize the changes in users' historical interest distribution $\mathbf{p}_u$ while ensuring the target group exposure distribution $\hat{\mathbf{q}}$; 2) the PCT-Reranker generates the final recommendation list according to personalized calibration targets $\hat{\mathbf{q}}_u$.

For example, to enhance platform reputation, the owner may identify a set of high-quality items (i.e., $\mathcal{G} = \{other, high\}$) and expect that these items take up half of the exposure resources (i.e., $\hat{\mathbf{q}} = [0.5, 0.5]$).

The system-level calibration is defined as follows:

Definition 2 (System-level Calibration). *The recommendation is system-level calibrated if the overall group exposure distribution is close to the target group exposure distribution.*

$$\mathbf{q} = \hat{\mathbf{q}}.$$

Similar disparity metrics as user-level calibration can be used to measure system-level calibration.

3.3 Tradeoff in Two-sided Calibration

User-level calibration and system-level calibration can hardly be achieved simultaneously. If the recommendation is perfectly user-level calibrated, the overall group exposure distribution will be:

$$\mathbf{q} = \frac{1}{|\mathcal{U}|}\sum_{u=1}^{|\mathcal{U}|}\mathbf{q}_u = \frac{1}{|\mathcal{U}|}\sum_{u=1}^{|\mathcal{U}|}\mathbf{p}_u = \mathbf{p}. \tag{3}$$

If the historical interest distribution on average (i.e., $\mathbf{p}$) does not equal the target group exposure distribution $\hat{\mathbf{q}}$ (the common case), the recommendation will not be perfectly system-level calibrated.

In the meantime, there are multiple ways to achieve perfect system-level calibration. For example, we can simply regulate each user's exposure distribution $\mathbf{q}_u$ to match the overall target $\hat{\mathbf{q}}$. However, this intuitive solution ignores the characteristics of each user, which may deviate from some users' historical interest distributions $\mathbf{p}_u$ to a large extent, yielding poor user-level calibration. This motivates us to find personalized calibration targets that minimize the changes in users' historical interest distributions while ensuring system-level calibration.

4 PERSONALIZED CALIBRATION TARGETS (PCT)

To better support the two-sided calibration task, we propose a post-processing method based on personalized calibration targets (PCT), as shown in Figure 2. The core idea is to solve personalized calibration targets $\hat{\mathbf{q}}_u$ that minimize the changes in users' historical interest distribution $\mathbf{p}_u$ while satisfying the overall target group exposure distribution $\hat{\mathbf{q}}$, which can be formulated as a linear programming problem. There are mainly two modules in PCT: 1) the PCT-Solver that solves personalized target exposure distributions $\hat{\mathbf{q}}_u$ and 2) the PCT-Reranker that reranks original recommendation results according to $\hat{\mathbf{q}}_u$. In the following, we will introduce these two modules in detail.

4.1 PCT Solver

Notice that perfect user-level calibration will make the overall group exposure distribution $\mathbf{q}$ consistent with the historical interest distribution on average $\mathbf{p}$, i.e., Eq.(3). Let $D(\cdot,\cdot)$ denote a disparity metric between two distributions (e.g., KL divergence, L2 distance). If we consider the current disparity $D(\mathbf{p},\hat{\mathbf{q}})$ between the averaged historical interest distribution $\mathbf{p}$ and the target group exposure distribution $\hat{\mathbf{q}}$ as a loss function, we can obtain the global gradient direction with respect to $\mathbf{p}$:

$$\mathbf{g} = \frac{\nabla_{\mathbf{p}} D(\mathbf{p},\hat{\mathbf{q}})}{||\nabla_{\mathbf{p}} D(\mathbf{p},\hat{\mathbf{q}})||}. \tag{4}$$

This gradient direction conveys the message about how to adjust users' exposure distributions to achieve system-level calibration on the basis of perfect user-level calibration.

Then, based on the historical interest distribution for each user $\mathbf{p}_u$, we can take the personalized calibrated target $\hat{\mathbf{q}}_u$ as the result of updating one step towards the negative gradient direction on top of $\mathbf{p}_u$:

$$\hat{\mathbf{q}}_u = \mathbf{p}_u - \gamma_u \mathbf{g}, \tag{5}$$

where γ_u is a scalar that represents the step size for each user. Then, the problem can be induced to solve these user-specific γ_u.

It is noteworthy that γ_u actually reflects the miscalibration degree for each user. If γ_u increases, the changes in the user's historical interest distribution $\mathbf{p}_u$ also enlarge, which leads to poorer user-level calibration. Therefore, we propose to minimize the weighted sum of γ_u with the constraint of ensuring 1) the system-level calibration target $\hat{\mathbf{q}}$ and 2) the legality of personalized calibration targets $\hat{\mathbf{q}}_u$, which can be formalized as a linear programming problem:

$$\begin{cases} \min\limits_{\gamma_u} & \sum_{u\in\mathcal{U}} w_u \gamma_u \\ \text{s.t.} & \sum_{u\in\mathcal{U}} \gamma_u \mathbf{g} = \sum_{u\in\mathcal{U}} (\mathbf{p}_u - \hat{\mathbf{q}}) \\ & 0 \le \gamma_u \le l_u,\ \forall u \in \mathcal{U}. \end{cases} \tag{6}$$

Here w_u measures the weight of each user ($w_u = 1$ by default). A larger weight will enforce more emphasis on this user to reduce the influence on his/her historical interest distribution, which can be determined by the application scenario (e.g., responding to users' adjustments). The first constraint makes sure that the target group exposure distribution can be achieved, i.e., $\sum_{u\in\mathcal{U}} \hat{\mathbf{q}}_u/|\mathcal{U}| = \hat{\mathbf{q}}$. The second constraint makes sure that the personalized calibration targets are legal distributions, i.e., $0 \le \hat{\mathbf{q}}_u \le 1$. To ensure such legality, the maximal step size l_u for each user can be calculated as:

$$l_u = \min\left(\left(\mathbf{p}_u(g) - I(\mathbf{g}(g) < 0)\right) / \mathbf{g}(g),\ g \in \mathcal{G} \right), \tag{7}$$

Based on the solved γ_u, we can obtain the personalized calibration targets $\hat{\mathbf{q}}_u$ via Eq.(5). Compared to the naive solution that lets $\hat{\mathbf{q}}_u = \hat{\mathbf{q}}$ (called Global-Solver hereafter), our PCT-Solver can also ensure that the system-level calibration target is achieved. Besides, minimizing γ_u helps avoid drastic changes to the historical interest distribution for each user, which benefits the user-level calibration. We will compare the distributions of user-level calibration of different methods in Section 5.4.

4.2 PCT Reranker

To achieve the personalized target group exposure distribution $\hat{\mathbf{q}}_u$ solved in PCT-Solver, we propose to balance the output obtained by a recommender system through reranking, which is a common practice in the literature [1, 29]. Here, a direct solution is to rerank each user's recommendation results based on maximum marginal relevance (MMR) [5]. This method starts with an empty recommendation set $\tilde{R}_u$ and iteratively adds items to it if adding that item makes the list more relevant and calibrated, until the size of $\tilde{R}_u$ reaches K [29].

In particular, the marginal relevance of each candidate item i can be defined as the combination of the predicted relevance $\hat{y}_{u,i}$ and the disparate exposure [19]:

$$s_{u,i} = \lambda \cdot \hat{y}_{u,i} - (1-\lambda) \cdot D\left(\hat{\mathbf{q}}_u, \mathbf{q}_{|\tilde{R}_u \cup \{i\}}\right). \tag{8}$$

Here $\mathbf{q}_{|\tilde{R}_u \cup \{i\}}$ is the group exposure distribution of the current item list $\tilde{R}_u$ together with the candidate item i. The tradeoff hyperparameter λ controls the balance between ranking performance and calibration. If $\lambda = 1$, the final recomemndation list $\tilde{R}_u$ is the same as the original R_u; otherwise $\tilde{R}_u$ is selected in consideration of disparate exposure.

Although the above MMR algorithm can obtain the reranking recommendation list $\tilde{R}_u$ as expected, it has two main limitations:

- MMR selects each item in a greedy approach, which lacks the global perspective and hence leads to sub-optimal results.
- The marginal relevance uses the original predicted score $\hat{y}_{u,i}$, while under the common pairwise learning setting, predicted scores of different users may have different ranges.

To alleviate the above limitations, our PCT-Reranker enhances the MMR algorithm in two folds. First, we "directly" add some originally top-ranked items to $\tilde{R}_u$ if this action will not exceed the target group exposure determined by the personalized calibration target $\hat{\mathbf{q}}_u$. Second, for the vacancies that no item can be added in the first iteration, we use the MMR algorithm to select the item with the maximal marginal relevance, where the predicted relevance score $\hat{y}_{u,i}$ is replaced with the rank of each item in R_u. The detailed algorithm is described in Algorithm 1.

In the first iteration, PCT-Reranker traverses all the top-K positions from 1 to K (Line 5-12). Notice that each position in the top-K recommendation list contributes a fixed exposure r_k according to its rank k. The total exposure resource in the top-K recommendation list is $\sum_{k=1}^{K} r_k$ (Line 1). Then, we can use the personalized calibration target $\hat{\mathbf{q}}_u$ to determine the target exposure resource allocated to different item groups (Line 2). For each position, we select the highest-ranked and unselected item in R_u that will not exceed the target exposure resource allocated to the corresponding item group (Line 7). When an item is selected at a specific position, the current exposure resource for each item group will be updated (Line 33). It is possible that no item can be selected for some specific position if the current exposure resource is close to the target one. These positions will be skipped temporally. The above pilot iteration directly selects some top-ranked items in R_u that are "safe" to appear in $\tilde{R}_u$. This enhances the global perspective of the traditional MMR algorithm because there is no need to ensure that every prefix of the recommendation list is calibrated.

Algorithm 1 PCT-Reranker Algorithm

Input: the number of items to recommend for each user K; original recommended item list R_u; ranking weight r_k; mapping from item to group $H(\cdot)$; personalized calibration targets $\hat{\mathbf{q}}_u$; tradeoff hyperparameter λ
Output: final recommended item list $\tilde{R}_u$

1: $resource \leftarrow r_1 + r_2 + \cdots + r_K$
2: $target \leftarrow \hat{\mathbf{q}}_u * resource$ # target exposure resource
3: $exp \leftarrow [0] * |\mathcal{G}|$ # current exposure resource
4: $\tilde{R}_u \leftarrow [-1] * K,\ selected \leftarrow set()$
5: **for** $k \leftarrow 1, 2, \cdots, K$ **do** # First Iteration
6: **for** $i \in R_u$ **do**
7: **if** $i \notin selected$ **and** $exp[H(i)] + r_k \leq target[H(i)]$ **then** # not exceed the target exposure resource
8: SELECT$(i,\ k)$
9: **break**
10: **end if**
11: **end for**
12: **end for**
13: **for** $k \leftarrow 1, 2, \cdots, K$ **if** $\tilde{R}_u[k] = -1$ **do** # Second Iteration
14: $score \leftarrow [-1] * |\mathcal{G}|,\ item \leftarrow [-1] * |\mathcal{G}|$
15: **for** $g \in \mathcal{G}$ **do** # score the top item for each group
16: $assume\_exp \leftarrow exp$
17: $assume\_exp[g] \leftarrow exp[g] + r_k$
18: $disparity \leftarrow (assume\_exp - target)^2.sum()/2$ # disparity if this item is added
19: **for** $rank, i \in enumerate(R_u)$ **do**
20: **if** $i \notin selected$ **and** $H(i) = g$ **then**
21: $score[g] \leftarrow \lambda/(rank + 1) - (1 - \lambda) * disparity$ # maginal relevance score
22: $item[g] \leftarrow i$
23: **break**
24: **end if**
25: **end for**
26: **end for**
27: SELECT$(item[argmax(score)],\ k)$
28: **end for**
29: **return** $\tilde{R}_u$
30:
31: **function** SELECT$(i,\ k)$ # select item i at position k
32: $\tilde{R}_u[k] \leftarrow i$
33: $exp[H(i)] \leftarrow exp[H(i)] + r_k$
34: $selected.add(i)$
35: **end function**

In the second iteration, PCT-Reranker selects items for the vacancies after the first iteration based on MMR (Line 13-28). We use the L2 distance as the default disparity measurement $D(\cdot, \cdot)$ (Line 18). Differently, to avoid wide-range scales of predicted scores, we leverage the rank of each item in R_u to calculate the marginal relevance (Line 21), which ranges from 0 to 1 for all the users. Besides, notice that if two items i, j belong to the same group and $\hat{y}_{u,i} > \hat{y}_{u,j}$, the marginal relevance score $s_{u,i} > s_{u,j}$ also holds because the disparity term only relies on the item group. As a result, we can only consider the highest-ranked item for each group and compare their marginal relevance scores (Line 27).

After the above two iterations, we can derive the reranked top-K recommendation list $\tilde{R}_u$ as expected. We will compare the proposed PCT-Reranker with traditional MMR in Section 5.3. Note that other advanced reranking methods [1, 27] can also be adopted here. We leave investigations of other rerankers as future work.

4.3 Efficiency Discussion

For the PCT-Solver, there have been mature algorithms [11] and toolkits (e.g., *scipy*, *gurobi*) to efficiently solve such an optimization problem. Besides, it is noteworthy that these personalized calibration targets can be solved in advance (offline), which is not the bottleneck of online services. Meanwhile, considering that the scale of this linear programming problem equals the number of users $|\mathcal{U}|$, the offline time complexity can also be non-negligible when the number of users grows very large. To accelerate this process, we propose to randomly split users into chunks with a maximum size $C < |\mathcal{U}|$. The chunk-level calibration target is directly set to the overall target $\hat{\mathbf{q}}$ on the whole, while personalized calibration targets are solved within each chunk. In this way, we can get $\lceil |\mathcal{U}|/C \rceil$ subproblems, and the system-level calibration is still ensured. These subproblems can be solved in parallel and the time complexity is controlled by the chunk size C. Although chunking users may not lead to the global optimum, our experiments show that we can obtain satisfactory results with a relatively small chunk size (e.g., $C = 5,000$), which enables the PCT-Solver to be applicable to large-scale scenarios.

For the PCT-Reranker, the time complexity is analyzed as follows. The first iteration takes $O(K|\mathcal{I}|)$ time in a naive way. However, this process can be accelerated to $O(K|\mathcal{G}|)$ with a similar idea to the merge sort. In particular, we can maintain an ordered list for each item group $g \in \mathcal{G}$, where all the items in the list belong to the same group and preserve the order in R_u. For each position, we compare the top-ranked items of $|\mathcal{G}|$ ordered lists. The item with the highest relevance score $\hat{y}_{u,i}$ and enough exposure resource for the corresponding group (adding this item will not exceed the target exposure) will be selected and then removed from the ordered list. In this way, the time complexity of the first iteration can be reduced to $O(K|\mathcal{G}|)$. In the second iteration, we will take $O(|\mathcal{I}|)$ to determine the highest-ranked items for each quality group. Then, the marginal relevance scores are computed in $O(|\mathcal{G}|^2)$. So the second iteration takes $O(K(|\mathcal{I}| + |\mathcal{G}|^2))$ time. The total time complexity is $O(K(|\mathcal{G}| + |\mathcal{I}| + |\mathcal{G}|^2))$. In comparison, ordinary MMR takes $O(K|\mathcal{I}||\mathcal{G}|)$. Considering that $|\mathcal{I}|$ is usually large and $K, |\mathcal{G}|$ are small, PCT-Reranker is much faster than MMR in practice. Our experiments will also compare the efficiency of different methods in Section 5.5.

5 EXPERIMENTS

In this section, we present our experimental settings and results. Our experiments are designed to answer the following research questions:

- **RQ1**: Can the proposed PCT method achieve better user-level calibration (and other objectives of recommendation) while ensuring system-level calibration compared to state-of-the-art baselines?
- **RQ2**: What are the impacts of the two main modules in PCT respectively (PCT-Solver and PCT-Reranker)?
- **RQ3**: Why are the personalized calibration targets solved in PCT-Solver helpful to enhance user-level calibration?
- **RQ4**: How about the efficiency of PCT compared to other reranking methods?

5.1 Experimental Settings

5.1.1 ***Datasets***. We use two datasets in our experiments, including both public and industrial datasets.

- **Tenrec** [38]: This is a recently published dataset collected from feeds recommendation platforms of Tencent, which includes four scenes split by the concrete platform and the item type. We use the QK-article subset because it contains the labeled quality of each item (article). Considering the whole subset is quite large, we randomly sample 20,000 users and filter out users/items with less than 5 associated interactions, yielding a dataset with 19,965 users, 31,413 items, and 884,315 interactions. Without loss of generality, we group items based on quality scores[2] and consider two groups $\mathcal{G} = \{normal, high\}$. Items whose quality scores are at least 7 are defined as the high-quality group (21% of the total item set), while the others are treated as the normal-quality group.
- **CMCC-Q**: This is an industrial dataset collected from China Mobile. This dataset contains video watching activities on smart TV, including 7,294 users, 1,971 videos, and 89,749 interactions. The quality of each video (binary scale, normal/high quality) is annotated according to three aspects: *production*, *culture value*, and *society value*. Videos are considered high-quality if they are both well-made and beneficial to the development of culture and society. Three experts are employed to annotate the quality of each video (Fleiss' kappa = 0.5889, reaching moderate agreement), and we use majority voting to get the final annotation. Finally, there are 183 high-quality items that take up 9.28% in the item set.

In the above two datasets, we use the quality attribute to group items because we are interested in improving the exposure of high-quality items (quality-aware responsible recommendation). Note that the proposed two-sided calibration can also work with other grouping attributes (e.g., provider, category, popularity) if the corresponding information is available and the recommendation platform cares about calibration towards those attributes.

5.1.2 ***Target Group Exposure Distribution.*** Our proposed two-sided calibration task supports a given target group exposure distribution $\hat{\mathbf{q}}$ for system-level calibration. Here we present two example policies that could be pursued. However, note that our proposed method is agnostic to the calibration target, which can be extended to any other policies depending on users, platform owners, or the law.

- **AvgEqual**: This policy aims to ensure that a group receives the exposure proportional to its ratio in the item set (average equity), i.e., $\hat{\mathbf{q}}(g) = |\mathcal{I}^g|/|\mathcal{I}|$.
- **Equal**: This policy aims to ensure the same degree of exposure among groups (overall equity), i.e., $\hat{\mathbf{q}}(g) = 1/|\mathcal{G}|$.

5.1.3 ***Backbone Recommenders.*** We use three different kinds of recommenders as the backbone to produce the original recommendation results:

- **BPRMF** [26]: This is a classic collaborative filtering method that optimizes MF with a pairwise ranking loss, where the negative item is randomly sampled from the item set.
- **LightGCN** [10]: This is a simplified graph convolution network for collaborative filtering that performs linear propagation between neighbors on the user-item bipartite graph.
- **SASRec** [13]: This is a typical sequential method that utilizes self-attention to exploit the mutual influence between historical interactions.

5.1.4 ***Evaluation Protocols.*** To support both general and sequential backbone models, we adopt the leave-one-out strategy to split the training, validation, and test set [16, 31, 33]. To evaluate the effect of two-sided calibration, we introduce the following two metrics to measure user-level calibration and system-level calibration, respectively:

[2]This dataset provides three quality scores by different scoring systems. We use item_score3 because it is the most fine-grained (10-level annotations) with reasonable distributions.

- **Miscalibration** (C_{KL}) [29]: This metric measures user-level calibration by the KL-divergence between users' historical interest distributions $\mathbf{p}_u$ and group exposure distributions $\mathbf{q}_u$:

$$C_{KL} = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{g \in \mathcal{G}} \mathbf{q}_u(g) \log \frac{\mathbf{q}_u(g)}{\mathbf{p}_u(g)}, \tag{9}$$

  which favors lower values for better user-level calibration.
- **Minority group exposure** (E_m) [19]: This metric measures system-level calibration by the overall exposure ratio of the minority group g_m (the high-quality group in our datasets, i.e., $g_m = high$):

$$E_m = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \mathbf{q}_u(g_m), \tag{10}$$

  which should be as close as the expected exposure $\hat{E}_m = \hat{\mathbf{q}}(g_m)$ determined by the system-level calibration target.

The minority group exposure E_m is intuitive and effective especially under our setting with two item groups because the exposure ratio of the other group will be $1 - E_m$. If E_m is close to the target value $\hat{E}_m$, we can say the system-level calibration is achieved.

For fair comparisons, we control the degree of system-level calibration of different methods and compare other metrics. In particular, we tune hyperparameters (e.g., the tradeoff hyperparameter λ in MMR) to make sure minority group exposure E_m is as expected ($\hat{E}_m \pm 0.01$). Besides the user-level calibration metric (C_{KL}), we also compare the ranking performance (NDCG) and the coverage of the minority group (COV_m). NDCG concerns whether the ground-truth item is ranked top in the recommendation list, while COV_m measures whether all the items in the minority group are recommended at least once.

5.1.5 ***Baselines.*** Note that previous calibrated recommendation studies cannot ensure the target group exposure distribution (system-level calibration). Thus, we only report the results of a typical calibrated recommendation method [29] for reference (**Calibrated**). We mainly compare our PCT with methods that are able to support the proposed two-sided calibration task.

- **Boosting**: This is a widely applied method in industry that gives higher scores to items in the minority group, i.e., $\hat{y}_{u,i} + \alpha \cdot I(H(i) = g_m)$. The target group exposure distribution can be achieved by tuning the weight α.
- **TFROM** [36]: This is a two-sided fairness-aware recommendation model that reranks the original recommendation results to control items' exposure and balance the loss of users' ranking performance.
- **RegExp** [19]: This is a recently proposed reranking method to regulate group exposure for items based on MMR, where each user's recommendation list is regulated towards the overall regulation target.

Note that some other related fairness-aware methods [4, 23, 28] have been encompassed in the above baselines. As for the previous quality-aware recommendation method [34], we find it can hardly achieve a given degree of exposure quality but only increases the exposure quality to some extent, which fails to achieve system-level calibration.

5.1.6 ***Implementation Details.*** We use the ReChorus [31] framework to run all the backbone recommendation models. For Calibrated, we set the tradeoff hyperparameter $\lambda = 0.5$ to balance user-level calibration and ranking performance. For other baselines, we tune corresponding hyperparameters to achieve similar E_m for fair comparisons. For PCT, we use the L2 distance as the disparity measurement to derive the gradient $\mathbf{g}$ in Eq.(4). To solve the linear programming problem, we use the *linprog* function in *scipy*. The chunk size C is set to 5000 because we find the

Table 1. Experimental results based on top-10 recommended lists on the Tenrec dataset. We control the degree of system-level calibration measured by the minority group exposure (E_m) and compare different methods in terms of ranking performance (NDCG), minority group coverage (COV_m), and user-level calibration (C_{KL}). ↑ means higher values are better and ↓ favors lower values. The best results among methods that achieve system-level calibration are in bold face, and the second best results are underlined. The * indicate $p \leq 0.05$ for the paired t-test of PCT vs. the best baseline (except for Calibrated).

Setting		Target Group Exposure Distribution $\hat{\mathbf{q}}$							
		AvgEqual ($\hat{E}_m = 0.21$)				Equal ($\hat{E}_m = 0.50$)			
Backbone	Method	E_m	NDCG ↑	COV_m ↑	C_{KL} ↓	E_m	NDCG ↑	COV_m ↑	C_{KL} ↓
BPRMF	Base	0.16	0.2379	0.3855	0.3858	0.16	0.2379	0.3855	0.3858
	+Calibrated	0.19	0.2335	0.4141	0.0519	0.19	0.2335	0.4141	0.0519
	+Boosting		**0.2375**	0.4463	0.2708		0.1929	0.6635	0.4505
	+TFROM	0.21	0.2316	0.4739	0.0530	0.50	0.2096	0.7108	0.2797
	+RegExp	±0.01	0.2270	0.4164	0.0532	±0.01	0.2105	0.6624	0.2566
	+PCT (ours)		0.2350	**0.5468***	**0.0081***		**0.2112**	**0.7560***	**0.2515***
LightGCN	Base	0.17	0.2725	0.6015	0.3400	0.17	0.2725	0.6015	0.3400
	+Calibrated	0.19	0.2670	0.6300	0.0692	0.19	0.2670	0.6300	0.0692
	+Boosting		**0.2716**	0.6644	0.2500		0.2204	0.7075	0.3952
	+TFROM	0.21	0.2650	0.6969	0.0526	0.50	0.2405	0.9119	0.2795
	+RegExp	±0.01	0.2587	0.6290	0.0532	±0.01	0.2409	0.8809	0.2575
	+PCT (ours)		0.2677	**0.7442***	**0.0089***		**0.2443***	**0.9307***	**0.2439***
SASRec	Base	0.17	0.2884	0.6594	0.3510	0.17	0.2884	0.6594	0.3510
	+Calibrated	0.19	0.2816	0.6726	0.0742	0.19	0.2816	0.6726	0.0742
	+Boosting		**0.2874**	0.7106	0.2666		0.2334	0.8948	0.3602
	+TFROM	0.21	0.2805	0.7438	0.0532	0.50	0.2591	0.9311	0.2796
	+RegExp	±0.01	0.2726	0.6819	0.0532	±0.01	0.2576	0.9070	0.2578
	+PCT (ours)		0.2825	**0.8010***	**0.0088***		**0.2633***	**0.9457***	**0.2433***

optimization problem of this size can be solved in seconds with satisfactory results. Each experiment is repeated 5 times with different random seeds and we report the average score. The codes are publicly available[3].

5.2 Overall Performance (RQ1)

Table 1 shows the reranking results based on top-10 recommendation lists on the public Tenrec dataset, integrated with various backbone models and target group exposure distributions (system-level calibration targets). From the experimental results, we mainly have the following observations.

Firstly, the typical calibrated recommendation method (Calibrated) only focuses on user-level calibration (lower C_{KL}) but fails to achieve system-level calibration (mismatch between E_m and $\hat{E}_m$). The calibrated results give higher minority group exposure E_m than the backbone model, but it can not approach the expected $\hat{E}_m$ according to the target group exposure distribution $\hat{\mathbf{q}}$. As a result, previous calibrated recommendation methods are still likely to result in undesired effects on the platform ecology. In Tenrec, the high-quality (minority) group takes up 21% in the item set but only receives 19% exposure even if the recommendation is user-level calibrated, which might not be expected by the platform owner. This validates the necessity of the proposed two-sided calibration task.

[3]https://github.com/THUwangcy/ReChorus/tree/RecSys23

Table 2. Experimental results on the CMCC-Q dataset. Other settings show similar results.

Setting		Target Group Exposure Distribution $\hat{\mathbf{q}}$			
		Equal ($\hat{E}_m = 0.50$)			
Backbone	Method	E_m	NDCG ↑	COV_m ↑	C_{KL} ↓
SASRec	Base	0.12	0.1848	0.5847	0.5708
	+Calibrated	0.15	0.1812	0.7213	0.1079
	+Boosting		0.1528	0.9071	0.6600
	+TFROM	0.50	0.1644	<u>0.9508</u>	0.4191
	+RegExp	±0.01	<u>0.1667</u>	<u>0.9508</u>	<u>0.3921</u>
	+PCT (ours)		**0.1763***	**0.9891***	**0.3838***

Secondly, other baseline methods can ensure the target group exposure distribution ($E_m \approx \hat{E}_m$), achieving the goal of system-level calibration. Comparing the two system-level calibration targets, the Equal policy expects higher minority group exposure and hence leads to a larger loss of user-level calibration (C_{KL}) and ranking performance (NDCG), which is more challenging than the AvgEqual policy. Different baselines also have different characteristics. Boosting is only effective to preserve ranking performance when the target exposure is close to the original one (AvgEqual), while generally suffering poor user-level calibration. TFROM and RegExp both take fairness into consideration, yielding similar NDCG and C_{KL}. The results show that although these fairness-aware methods do not directly optimize user-level calibration, they can bring better-calibrated results than the naive Boosting method. In comparison, TFROM better improves the minority group coverage (higher COV_m) due to the specially designed reranking strategy. RegExp gives better user-level calibration under the Equal target, which may benefit from treating all the users equally.

Last but not the least, the proposed PCT achieves significantly better user-level calibration than other methods while ensuring system-level calibration. Besides, PCT can maintain competitive ranking performance and improve minority group coverage to a large extent. In particular, PCT yields extremely low C_{KL} under the AvgEqual policy. The proposed PCT-Solver help avoid drastic changes in users' historical interest distributions while ensuring the target group exposure distribution. Note that although the NDCG and C_{KL} are worse than Calibrated under the Equal policy, they are not comparable because Calibrated does not achieve system-level calibration. PCT is still the best among baselines that achieve similar E_m. Furthermore, Table 2 shows the experimental results on the CMCC-Q dataset. Here we mainly show the results when integrated with the more challenging Equal policy and the most powerful backbone SASRec (other settings yield similar results). The consistent superior performance compared to other baselines validates the effectiveness of PCT.

5.3 Ablation Study (RQ2)

There are two main modules in PCT, namely PCT-Solver and PCT-Reranker. In this section, we replace each module with other methods to show the impacts of PCT-Solver and PCT-Reranker respectively. On the one hand, when determining personalized calibration targets $\hat{\mathbf{q}}_u$, we replace the proposed PCT-Solver with **Global-Solver**, which simply lets $\hat{\mathbf{q}}_u = \hat{\mathbf{q}}$. Each user's recommendation list will be calibrated towards the overall target group exposure distribution. On the other hand, when reranking each user's recommendation results according to $\hat{\mathbf{q}}_u$, we replace the proposed PCT-Reranker with **MMR**, which iteratively adds items that make the list more relevant and calibrated in a greedy way.

Table 3. Ablation Study on the public Tenrec dataset. We replace the solver and the reranker in PCT.

Method	Setting		AvgEqual ($\hat{E}_m = 0.21$)			
	Solver	Reranker	E_m	NDCG ↑	COV_m ↑	C_{KL} ↓
BPRMF	-	-	0.16	0.2379	0.3855	0.3858
(a)	Global-Solver	MMR		0.2281	0.4150	0.0533
(b)	PCT-Solver	MMR	0.21	0.2289	0.5170	0.0132
(c)	Global-Solver	PCT-Reranker	±0.01	0.2317	0.4831	0.0525
PCT	PCT-Solver	PCT-Reranker		**0.2350**	**0.5468***	**0.0081***

Table 3 shows the results when integrated with the AvgEqual target and the BPRMF backbone on the Tenrec dataset (other settings yield similar results). First, comparing (b) vs. (a) and PCT vs. (c), we can see that the PCT-Solver is always better than the Global-Solver. The PCT-Solver not only significantly improves the user-level calibration C_{KL} but also leads to higher NDCG and COV_m. Second, comparing (c) vs. (a) and PCT vs. (b), the PCT-Reranker can further benefit the improvements of NDCG and COV_m, as well as slightly enhance C_{KL}. As a result, both PCT-Solver and PCT-Reranker are of great importance, which helps the full PCT achieve the best results on the whole.

5.4 Analysis on User-level Calibration (RQ3)

To further understand the rationale of the proposed PCT-Solver, Figure 3 shows the distributions of user-level calibration $KL(\mathbf{p}_u \,||\, \mathbf{q}_u)$ of different methods, integrated with the Equal target and the BPRMF backbone on the Tenrec dataset. The main experiments in Section 5.2 have shown that PCT is superior to other methods in terms of the overall C_{KL}, while Figure 3 gives a more fine-grained perspective about user-level calibration. It is clear to see that Boosting could bring a strong mismatch between the historical interest distribution $\mathbf{p}_u$ and group exposure distribution $\mathbf{q}_u$ for some users. RegExp alleviates this issue to some extent but there still exist users will high KL-divergence. Differently, the PCT-Solver in our PCT method seeks to minimize the changes in users' historical interest distributions, and hence the maximal KL-divergence is significantly smaller than other methods. This conveys the message that no user's interest distribution is greatly altered in the recommendation list generated by PCT. Besides, notice that the distribution of KL-divergence is much more centered than other methods. As a result, PCT is capable of achieving better user-level calibration while ensuring system-level calibration, which distributes the distribution shift to all the users more equally.

5.5 Efficiency Comparison (RQ4)

Finally, we show the reranking efficiency of different methods on the Tenrec dataset. For fair comparisons, The efficiency experiments are conducted on the same machine (Intel Core 12-core CPU of 3.5GHz). We can see that PCT is a little slower than TFROM but much faster than RegExp. Each user's recommendation list can be reranked in 1.5ms by PCT, which is generally acceptable in practice. Compared to RegExp which uses the conventional MMR for reranking, PCT does not need to calculate the marginal relevance score of each candidate item and hence leads to lower time complexity as discussed in Section 4.3.

6 CONCLUSION

In this paper, we investigate the two-sided calibration task for quality-aware responsible recommendation. Previous calibrated recommendation studies only focus on user-level calibration. However, with the increasing calls for responsible

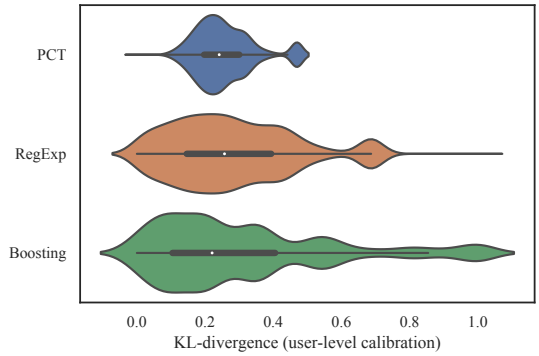


Fig. 3. Distributions of KL-divergence between a user's historical interest distribution $\mathbf{p}_u$ and group exposure distribution $\mathbf{q}_u$.

Method	TFROM	RegExp	PCT
Reranking Time	14sec	22min	30sec

Table 4. Efficiency of reranking 19,965 users' top-10 recommended item lists on the Tenrec dataset.

recommendation, it has become inadequate to just match users' past interest distributions, which could still lead to undesired effects on the platform ecology. To this end, we propose the two-sided calibration task that additionally pursues system-level calibration, where the target group exposure distribution could be determined by users, platform owners, and even the law. To better support the two-sided calibration task, we propose a post-processing method based on personalized calibration targets (PCT). For one thing, PCT-Solver solves personalized target group exposure distributions to minimize the changes in users' historical interest distributions while ensuring the overall target exposure distribution. For another, PCT-Reranker reranks the original recommendation results to generate both relevant and calibrated recommendation lists. Experiments on public and industrial datasets show that PCT can achieve better user-level calibration while satisfying system-level calibration than state-of-the-art baselines. PCT also achieves comparative ranking performance and significantly improves coverage of the minority item group.

In the future, we plan to validate the long-term benefits of our method via online experiments. It is also interesting to investigate the connection between fairness and calibration. Our experiments show that they might benefit from each other, but existing methods mainly focus to address one of them.

REFERENCES

[1] Himan Abdollahpouri, Zahra Nazari, Alex Gain, Clay Gibson, Maria Dimakopoulou, Jesse Anderton, Benjamin Carterette, Mounia Lalmas, and Tony Jebara. 2023. Calibrated Recommendations as a Minimum-Cost Flow Problem. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*. 571–579.

[2] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H Chi, et al. 2019. Fairness in recommendation ranking through pairwise comparisons. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2212–2220.

[3] Alex Beutel, Ed H Chi, Fernando Diaz, and Robin Burke. 2020. Responsible recommendation and search systems. *WWW 2020* (2020).

[4] Asia J Biega, Krishna P Gummadi, and Gerhard Weikum. 2018. Equity of attention: Amortizing individual fairness in rankings. In *The 41st international acm sigir conference on research & development in information retrieval*. 405–414.

[5] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. 335–336.

[6] Pablo Castells, Neil Hurley, and Saul Vargas. 2022. Novelty and diversity in recommender systems. In *Recommender systems handbook*. Springer, 603–646.

[7] L Elisa Celis, Sayash Kapoor, Farnood Salehi, and Nisheeth Vishnoi. 2019. Controlling polarization in personalization: An algorithmic framework. In *Proceedings of the conference on fairness, accountability, and transparency*. 160–169.

[8] Mehdi Elahi, Dietmar Jannach, Lars Skjærven, Erik Knudsen, Helle Sjøvaag, Kristian Tolonen, Øyvind Holmstad, Igor Pipkin, Eivind Throndsen, Agnes Stenbom, et al. 2021. Towards responsible media recommendation. *AI and Ethics* (2021), 1–12.

[9] Miriam Fernández, Alejandro Bellogín, and Iván Cantador. 2021. Analysing the effect of recommendation algorithms on the amplification of misinformation. *arXiv preprint arXiv:2103.14748* (2021).

[10] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 639–648.

[11] Qi Huangfu and JA Julian Hall. 2018. Parallelizing the dual revised simplex method. *Mathematical Programming Computation* 10, 1 (2018), 119–142.

[12] Kojiro Iizuka, Yoshifumi Seki, and Makoto P Kato. 2021. The Effect of News Article Quality on Ad Consumption. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 3107–3111.

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

[14] Hang Lei, Yin Zhao, and Longjun Cai. 2020. Multi-objective optimization for guaranteed delivery in video service platform. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 3017–3025.

[15] Jurek Leonhardt, Avishek Anand, and Megha Khosla. 2018. User fairness in recommender systems. In *Companion Proceedings of the The Web Conference 2018*. 101–102.

[16] Jiacheng Li, Yujie Wang, and Julian McAuley. 2020. Time Interval Aware Self-Attention for Sequential Recommendation. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 322–330.

[17] Yunqi Li, Hanxiong Chen, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2021. User-oriented fairness in recommendation. In *Proceedings of the Web Conference 2021*. 624–632.

[18] Hongyu Lu, Min Zhang, Weizhi Ma, Yunqiu Shao, Yiqun Liu, and Shaoping Ma. 2019. Quality effects on user preferences and behaviorsin mobile news streaming. In *The World Wide Web Conference*. 1187–1197.

[19] Mirko Marras, Ludovico Boratto, Guilherme Ramos, and Gianni Fenu. 2022. Regulating Group Exposure for Item Providers in Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1839–1843.

[20] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. 2020. Controlling fairness and bias in dynamic learning-to-rank. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 429–438.

[21] Mohammadmehdi Naghiaei, Hossein A Rahmani, Mohammad Aliannejadi, and Nasim Sonboli. 2022. Towards confidence-aware calibrated recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 4344–4348.

[22] Jinoh Oh, Sun Park, Hwanjo Yu, Min Song, and Seung-Taek Park. 2011. Novel recommendation based on personal popularity tendency. In *2011 IEEE 11th international conference on data mining*. IEEE, 507–516.

[23] Gourab K Patro, Arpita Biswas, Niloy Ganguly, Krishna P Gummadi, and Abhijnan Chakraborty. 2020. Fairrec: Two-sided fairness for personalized recommendations in two-sided platforms. In *Proceedings of The Web Conference 2020*. 1194–1204.

[24] Martin Potthast, Sebastian Köpsel, Benno Stein, and Matthias Hagen. 2016. Clickbait detection. In *European conference on information retrieval*. Springer, 810–817.

[25] Bashir Rastegarpanah, Krishna P Gummadi, and Mark Crovella. 2019. Fighting fire with fire: Using antidote data to improve polarization and fairness of recommender systems. In *Proceedings of the twelfth ACM international conference on web search and data mining*. 231–239.

[26] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th conference on uncertainty in artificial intelligence*. AUAI Press, 452–461.

[27] Sinan Seymen, Himan Abdollahpouri, and Edward C Malthouse. 2021. A constrained optimization approach for calibrated recommendations. In *Proceedings of the 15th ACM Conference on Recommender Systems*. 607–612.

[28] Ashudeep Singh and Thorsten Joachims. 2018. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2219–2228.

[29] Harald Steck. 2018. Calibrated recommendations. In *Proceedings of the 12th ACM conference on recommender systems*. 154–162.

[30] Saúl Vargas and Pablo Castells. 2011. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*. 109–116.

[31] Chenyang Wang, Weizhi Ma, Min Zhang, Yiqun Liu, and Shaoping Ma. 2020. Make It a Chrous: Knowledge- and Time-aware Item Modeling for Sequential Recommendation. In *Proceedings of the 43th International ACM SIGIR conference*. ACM.

[32] Chenyang Wang, Yuanqing Yu, Weizhi Ma, Min Zhang, Chong Chen, Yiqun Liu, and Shaoping Ma. 2022. Towards Representation Alignment and Uniformity in Collaborative Filtering. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1816–1825.

[33] Chenyang Wang, Min Zhang, Weizhi Ma, Yiqun Liu, and Shaoping Ma. 2019. Modeling Item-Specific Temporal Dynamics of Repeat Consumption for Recommender Systems. In *The World Wide Web Conference*. ACM, 1977–1987.

[34] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2022. Quality-aware News Recommendation. *arXiv preprint arXiv:2202.13605* (2022).

[35] Haolun Wu, Bhaskar Mitra, Chen Ma, Fernando Diaz, and Xue Liu. 2022. Joint multisided exposure fairness for recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 703–714.

[36] Yao Wu, Jian Cao, Guandong Xu, and Yudong Tan. 2021. Tfrom: A two-sided fairness-aware recommendation model for both customers and providers. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1013–1022.

[37] Xing Yi, Liangjie Hong, Erheng Zhong, Nanthan Nan Liu, and Suju Rajan. 2014. Beyond clicks: dwell time for personalization. In *Proceedings of the 8th ACM Conference on Recommender systems*. 113–120.

[38] Guanghu Yuan, Fajie Yuan, Yudong Li, Beibei Kong, Shujie Li, Lei Chen, Min Yang, Chenyun Yu, Bo Hu, Zang Li, et al. 2022. Tenrec: A Large-scale Multipurpose Benchmark Dataset for Recommender Systems. *Advances in Neural Information Processing Systems* 35 (2022), 11480–11493.