Universiteit Leiden

# Master Computer Science

A Reproducibility Study on Recommender Systems and Fairness Aware Models: Accuracy and Fairness Trade Off


| | |
|---|---|
| Name: | Seyedehsheida Nehzati |
| Student ID: | s3626962 |
| Date: | 23/01/2026 |
| Specialisation: | Data Science |
| 1st supervisor: | ZhaoChun Ren |
| 2nd supervisor: | QinYu Chen |


Master's Thesis in Computer Science


Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

# Contents

# Chapter 1

# Introduction

Recommender system have become a widely used models in digital platforms that help users in browsing products, movies, music and new. Recommender systems reduce information overload on users,by leaning historical interaction in the dataset and recommending a relevant item to the user accordingly.[30, 13]. While Recommender models have been critical in user engagement and retention, they can also perform as a socio-technical operator that decides which user sees which item. [14, 21]. As these decision are repeatedly made over time, it can lean to a systematic bias that has consequences for both user, item provides and the platform itself. [21, 12].

Fairness in recommender systems is about different individuals or groups being treated equally by the recommendation process or its outcomes [21]. Fairness can be defined as either user fairness, item (provide) fairness, or joint fairness. User fairness can be defined as whether or not different users receive same quality of recommendations, while item or provider fairness can be defined as whether different items or providers receive enough exposure, and lastly joint fairness can be described as balancing fairness for both user and item.[33].

A common leading cause of unfair recommendation is popularity bias. Popularity bias, is the tendency of recommenders to over recommend a popular items (short-head) and under-recommend less popular items (long-tail)[33]. Typically logged interaction data, have a heavy-tailed distribution where a small fraction of items account for most interactions. In this scenario, if models optimize accuracy and their recommendation on

these data, they learn to favor head items due to their popularity. In such cases, even when a user genuinely likes a minority or tail item, such items receive disproportionately low exposure [21].

As a consequence, for users, over-emphasising on popular items can reduce diversity, novelty, and serendipity of recommendation lists, which reinforces filter bubbles and limits the exposure to niche contents [33]. Item providers, especially those with low exposure, get fewer opportunities to receive interactions and potential income[3, 12]. Over time, this can create a loop of Matthew effect, where the rich get richer and poor gets poorer.

In response, many recent studies in recommender systems have been focusing on fairness topic. The existing methods can be categorized as following:

*Pre-Processing methods* modify training data prior to the training, aiming to reduce imbalance between groups. Some methods re-samples users or items so that disadvantaged minority groups are better represented, while others add antidote data to counteract popularity biases [28, 11].

*In-processing (ranking) methods* change the learning objective or architecture to directly include fairness goal within the recommendation task. Some methods in this category, add regularization terms that penalize disparities in exposure between groups, ensuring that score distributions are less correlated with sensitive attributes, or learning fair representations via adversarial objectives [41, 43, 7]. These approaches can target either user-side or item-side fairness. However, although these are flexible in principle, but they require model modification and can be costly to tune.

*Post-processing (re-ranking) methods* adjust the ranked lists produced by a base recommender to satisfy fairness goals. Many such methods implement constrained or multi-objective re-ranking to balance utility with exposure constraints for minority or disadvantage groups [32, 26]. A benefit of re-ranking is that it operates close to the final presentation shown to users and can be plugged on top of existing recommenders without retraining. However, it is limited by the candidate set and potential biases that has already been seeded in base scores [21].

Across these categories, a common challenge is the trade-off between fairness and accuracy. Fairness interventions often improve exposure or parity metrics for disadvantaged groups at the cost of reducing standard utility metrics like Recall@$k$ or

NDCG@$k$ [41, 43, 21]. Studies has shown that stronger fairness constraints imply accuracy loss. This raises practical questions: How much accuracy must one sacrifice for a given gain in fairness? Are some methods more likely to lose accuracy for fairness gain? Does this trade-off behave consistently across datasets and base recommenders?

Existing fairness-aware methods demonstrate that it is possible to reduce popularity-driven under-recommendation and improve exposure parity, yet several gaps remain. First, evaluation practices are heterogeneous. Different papers use different datasets, base recommendation models (e.g., matrix factorization, neural collaborative filtering, sequential models), and fairness definitions, making results hard to compare and synthesize [21]. Some works target item popularity groups, others focus on provider categories or demographic attributes; some measure fairness on exposure, others on calibration or equal opportunity [21]. Without a unified benchmark, it is difficult to answer basic comparative questions such as: Does an in-processing method generally perform better than a re-ranking method on the same backbone? Are conclusions stable across domains like movies, music, and e-commerce?

Second, most studies fix a single base recommender and examine one family of fairness interventions, rather than systematically varying both the base model and the fairness approach. For example, item under-recommendation bias has mostly been studied on BPR-style matrix factorization [43], while two-sided fairness re-ranking has typically been evaluated with specific collaborative filtering baselines [24]. It remains unclear how these fairness-aware models interact with modern sequential architectures.

Third, while many papers qualitatively acknowledge an accuracy–fairness trade-off, we lack systematic, quantitative mapping of the trade-off curves across methods and datasets [41, 21, 38, 39]. In particular, little is known about how sensitive different methods are to hyperparameter changes that strengthen or relax fairness constraints, and whether methods that look similar at one operating point behave differently when we sweep fairness strength.

These gaps motivate a systematic study of fairness–accuracy trade-offs under a unified experimental design. Specifically, we aim to explore how representative fairness-aware models from different families perform when combined with different base recommenders and applied to multiple standard benchmarks, and to interpret their trade-offs through a common lens.

To keep the study focused yet representative, we adopt the following scope. As base

recommenders, we select: a sequential model (SASRec) that captures user interaction dynamics and reflects current practice in personalized recommendation [18]; and a standard implicit-feedback collaborative filtering baseline (e.g., BPR-style matrix factorization [29]).

On top of these, we consider fairness-aware methods from two main families: 1) In-processing methods, such as a debiased personalized ranking model that targets item under-recommendation bias [41], and an approach that enforces low dependence between recommendations and a group attribute[17]. 2) Post-processing (re-ranking) methods, including a provider max-min fairness re-ranker and a two-sided fairness re-ranker that jointly optimize consumer and producer fairness [36][24], as well as ElasticRank, a fair re-ranking algorithm that employs elasticity calculations to adjust inter-item distances within a curved space [39].

We evaluate all combinations on widely used public datasets that span different domains and short-head-long-tail popularity bias, such as MovieLens-100K and Steam. These datasets offer explicit or implicit feedback and allow us to construct popularity-based item groups (e.g., head vs. tail) for fairness analysis.

Guided by the above motivation, this thesis addresses the following research questions:

1. **RQ1 (Trade-off characterization).** Under a unified pipeline and evaluation protocol, how do different fairness-aware methods (in-processing vs. post-processing) shift the fairness–accuracy trade-off? In particular, how do they affect item popularity-based fairness metrics (e.g., tail exposure) versus standard utility metrics (NDCG@$k$)?

2. **RQ2 (Robustness across models and datasets).** How consistent are these trade-offs across different base recommenders (e.g., SASRec vs. a BPR-style model) and across datasets from different domains (MovieLens, Steam)? Do some fairness methods generalize better across domains or architectures?

The remainder of this thesis is organized as follows. Chapter 2 reviews background and related work on recommender systems, bias and fairness in recommendation and existing fairness-aware methods. Chapter 3 contains the methodology, and the pipeline of this study, and describes the datasets and statistical analysis on the chosen datasets. Chapter 4 presents the results on fairness and accuracy for all combinations of method–model-dataset. In chapter 5, we discuss our experimental result and finally

in Chapter 6 we conclude the accuracy-fairness trade-off.

# Chapter 2

# Literature Review

## 2.1 Fairness and Bias in Recommender Systems

Recommender systems are now embedded in many online platforms and influence which movies we watch, which products we buy, and even which jobs we see [28]. Because they allocate information and exposure, they do not only optimize user satisfaction but also affect social and economic opportunities for users and item providers [23]. Recent work has shown that recommender systems can behave unfairly. For example, different user groups can receive very different recommendation quality, and minority or "long-tail" items can systematically get less exposure than popular ones [9]. On the user side, studies report that women or older users may receive worse recommendations than other demographic groups. On the item side, long-tail or minority items tend to be under-recommended, and popular items receive disproportionate exposure[2, 6].

A key concept related to fairness is bias. Bias describes systematic statistical distortions between the data or the model output and the "true" underlying reality, such as popularity bias, position bias, and selection bias[20, 5]. Fairness, in contrast, asks whether the allocation of recommendations should be considered just or acceptable for different stakeholders, such as users, items, or providers. Because recommender systems are data-driven, they can pick up and amplify existing data biases, which then translate into unfair outcomes if not carefully controlled. This motivates the growing

research area of fairness-aware recommender systems [22].

## 2.2 Fairness Definitions in Recommender Systems

Fairness in recommender systems is more complex than in classical classification because there are multiple stakeholders (users, items, providers) and because recommendations are dynamic and personalized. Surveys such as Wang et al. and Li et al. [33, 21] highlight several important aspects of fairness.

Fairness in recommender systems can be defined as either Process or Outcome fairness. Process fairness asks whether the model's internal processing uses sensitive attributes in a fair way, e.g., whether user or item representations are independent of protected attribute[19]. On the other hand Outcome fairness focuses on the fairness of the final results, such as the exposure, ranking positions, or click-through rates different groups receive [10].

Another important axis is whether fairness is defined at the individual or group level. Individual fairness requires that similar individuals receive similar treatment. In recommendation, this might mean that two similar users should see similar-quality lists, or that similar items should receive similar prediction errors. Group fairness requires that defined groups (e.g., popular vs. tail items, male vs. female users) are treated fairly on average. Existing work shows that most fairness methods in recommendation target group fairness rather than individual fairness, because it is easier to define and optimize group-level metrics [33].

Fairness can be defined for different "subjects", such as item, user or provider. While user fairness addresses different user groups equal treatment in terms of accuracy, diversity, item fairness addresses how different item groups (e.g., long-tail vs. head items, or items from different providers) receive fair exposure or recommendation quality. Many works focus on either user fairness or item/provider fairness alone, and recent surveys point out that joint fairness is challenging because improving fairness for one side can hurt the other[1, 4].

This thesis focuses on item-side group fairness under popularity-based groupings (head vs. tail items) and measures whether exposure is fairly distributed between these groups, while still preserving recommendation utility.

## 2.3 Fairness aware Models

Research on fairness-aware recommender systems has expanded over a few broad paradigms for intervention, typically distinguished by where they act in the recommendation pipeline: at the data level (pre-processing), inside the learning algorithm (in-processing), or over the final ranked list (post-hoc re-ranking). Across these paradigms, methods aim to reduce disparities between protected and unprotected groups, whether defined on the user side (gender, age), the item or provider side (popularity, minority businesses) or jointly for multiple stakeholders.[8] [15]

### 2.3.1 Pre-processing methods

Data-oriented or pre-processing methods operate before model training by modifying the input data such that the recommender model sees a less biased distribution. A straightforward strategy is resampling: Ekstrand and colleagues adjust the proportion of protected users in the training set to alleviate demographic disparities, showing that such shifts can reduce unfairness but often only modestly [9]. Other work augments the data with "antidote" samples specifically crafted to counteract observed biases, for example by adding synthetic ratings that improve both polarization and fairness in collaborative filtering [28]. In practice, these methods have the advantage of being model-agnostic, any recommender can be trained on the adjusted data, but their effect can be effected by later components in the pipeline, especially ranking and re-ranking stages that are not fairness-aware.

### 2.3.2 In-processing methods

In-processing methods aim to embed fairness into the model representations during training. One way to achieve this is through Regularization (Reg), where a quantifiable penalty term is added to the standard recommendation loss function. This penalty enforces recommendation independence, aiming to ensure that the predicted preference score is statistically independent of a sensitive attribute $S$, such as an items popularity group (head or tail), thereby mitigating the influence of unwanted, biased information like popularity during the core prediction stage [17]. The second form of in-processing intervention is based on penalizing prediction errors differentially across groups, as proposed in the work by Yao and Huang[41]. This method, addresses fairness by integrating specific unfairness metrics, such as Value Unfairness, or Under-prediction Unfairness, directly into the learning objective. By defining the penalty based on the

difference in average estimated error or prediction direction between disadvantaged and advantaged groups, the training process is explicitly guided to reduce the disparity in prediction quality, ensuring, for example, that one group is not consistently underestimated while another is overestimated. This mechanism forces the model to achieve fairer prediction quality across groups directly through the learning objective [41]

A complementary line of in-processing work uses adversarial learning to enforce process fairness by filtering out sensitive information from latent representations. In these approaches, a discriminator is trained to predict sensitive attributes (such as gender or item group) from user or item embeddings, while the main model is trained to both predict relevance and "fool" the discriminator, pushing the embeddings toward invariance with respect to the protected attribute[34, 43]. Such adversarial strategies have been applied in graph-based recommendation, news recommendation, and user modeling with missing sensitive attributes, and can target either representations or predicted scores.

### 2.3.3 Post-processing methods

The third category of fairness aware models is Post-processing re-ranking methods, that focus on adjusting the output ranking produced by a base recommender, using the base model's scores as inputs. Among these methods, FairRec is a Two-sided and provider-oriented reranking model. FairRec maps the provider–consumer fairness problem in two-sided platforms that constructs recommendation lists that guarantee each producer a minimum "maximin share" of exposure, and at the same time ensuring envy-free up to one item fairness for users[27]. For provider fairness under dynamic user traffic, BankFair introduces a bankruptcy-inspired model. BankFair treats daily exposure as a resource to be allocated under minimum exposure requirements and uses the Talmud rule to shift fairness intensity between high-traffic and low-traffic periods so that short-term accuracy remains acceptable while long-term provider fairness is maintained[42].

FairSync, on the other hand, approaches fairness problem by solving a constrained distributed optimization problem in dual space which aggregates historical fairness information and configures item embeddings so that distributed nearest-neighbor search respects group-level minimum exposure constraints while preserving recall and NDCG [37].

P-MMF (Provider Max-Min Fairness Re-ranking) focuses its objective on the Rawlsian Maximin Fairness (MMF), prioritizing the optimization of the utility of the worst-off item group to improve stability and ensure minimum market share across all providers [35]. Extending this focus to a multi-stakeholder setting, CPFair, Personalized Consumer and Producer Fairness Re-ranking, is designed for two-sided platforms. CP-Fair aims for a simultaneous balance between Consumer Fairness (user) and Producer Fairness (item exposure) by formulating the task as an optimization problem[25]. A different approach is offered by ElasticRank, which analyzes the Accuracy-Fairness Trade-off through the lens of economic elasticity. ElasticRank likens the fairness constraint to a commodity tax whose cost is transferred to users as accuracy loss; it seeks to minimize this transfer by strategically adjusting inter-item distances in the ranking space, prioritizing fairness interventions where the resulting accuracy cost is lowest [40].

## 2.4 Base Recommender models

Most fairness-aware recommender-system studies are built on top of standard base models, chosen because they are well understood, widely used in the recommendation literature, and representative of different modeling paradigms. Surveys of bias and fairness in recommendation consistently report that experimental work typically relies on a mix of simple popularity-based baselines, latent-factor collaborative filtering, neighbourhood methods, deep models, and sequential recommenders, and then layers fairness mechanisms on top of these base models.

A large fraction of group-fairness methods are built on top of matrix factorization. For example, in Fair Recommendations with Limited Sensitive Attributes, Shi et al. use a basic matrix factorization (MF) model as the backbone for all methods, and then add a demographic-parity regularizer or distributionally robust optimization term to the MF loss to obtain fair outcome[31]. Similarly, many regularization-based approaches summarized in the survey by Wang et al. are explicitly described as “add fairness regularization to SLIM” or “fairness-aware tensor-based recommendation,” meaning that fairness terms are attached to existing MF- or tensor-factorization models rather than entirely new architectures [33]

Pairwise ranking models, especially Bayesian Personalized Ranking (BPR), are also widely adopted as base models in fairness and debiasing work. The original BPR paper

formulates personalized ranking from implicit feedback as a pairwise maximum-posterior problem and shows how to optimize models such as MF and k-nearest-neighbor (kNN) directly for ranking quality using stochastic gradient descent on sampled triplets $(u, i, j)$ [29]. Building on this, several recent fairness methods assume a BPR-trained base model and then intervene at the sampling or re-ranking stage. FairNeg, for instance, proposes a "fairly adaptive negative sampling" scheme to improve item-group fairness, explicitly stating that it is implemented on top of two representative recommendation models, MF and LightGCN, both trained with the BPR loss [7]. TaxRank, a taxation-inspired fair re-ranking framework, uses a BPR model to compute click-through-rate (CTR) estimates for each user–item pair and then redistributes exposure based on these scores, again treating BPR as the underlying ranking engine whose outputs are adjusted for fairness [38].

Sequential and interactive models are less common but increasingly important base architectures in fairness-related work. The Self-Attentive Sequential Recommendation model (SASRec), while not explicitly a fairness mechanism, utilizes a Transformer-style self-attention mechanism to efficiently process a user's interaction sequence in parallel. This architecture enables it to adaptively weigh the relevance of past actions, successfully capturing long-term sequential semantics. studies confirm that SASRec significantly outperforms earlier sequential approaches like RNN/CNN-based models in terms of both accuracy (Hit Rate and NDCG) and computational efficiency, making it a powerful modern backbone for next-item prediction [18].

Most foundational and recent fairness studies prioritize applying interventions to matrix factorization models (often optimized with BPR), or adapting deep and graph-based collaborative filtering methods such as NeuMF, VAE-based CF, and LightGCN, rather than to modern self-attentive sequential architectures. Representative examples include FairNeg (MF and LightGCN with BPR), DRFO (MF backbone), CPFair (PF, WMF, NeuMF, VAECF), HetroFair and FA4GCF (GNN-based fairness), as well as the survey evidence in Wang et al., Deldjoo et al., and Jin et al. [7, 27, 31, 25, 33, 15].

Few studies address fairness within the context of contemporary self-attentive sequential models. Therefore, integrating SASRec into this thesis serves a dual purpose: first, it ensures our utility measurements are derived from a sequential architecture, and second, its inclusion directly addresses a gap in current fairness research by extending the scope of critical benchmarking beyond traditional static models into modern, high-performance sequential recommendation paradigms.

## 2.5 Summary

Overall, the literature paints a rich but fragmented picture. Data-level, in-processing, and post-hoc interventions have each been shown to reduce particular types of unfairness under specific conditions, yet there is no consensus on which paradigm or theoretical foundation yields the most favorable accuracy–fairness trade-offs across models and domains. Existing surveys highlight that this difficulty stems from inconsistent evaluation practices: studies commonly rely on different datasets, adopt varying fairness definitions, and employ heterogeneous metrics, which hampers clear generalization. Moreover, comparisons are often confined to methods within the same family—re-ranking techniques are usually pitted against other re-ranking methods, and regularization approaches are compared only with other regularizers—so cross-paradigm insights remain scarce. While the broader fairness literature utilizes a wide array of recommender backbones, systematic benchmarking across fundamentally different learning approaches is limited, especially when it comes to modern sequential models. Additionally, much of the recent work concentrates on provider-fairness objectives and pays comparatively little attention to group fairness defined by the head–tail split of items, which reflects a core popularity bias in many recommendation settings.

These observations motivate the present thesis. By selecting a small but representative set of methods—max-min based provider re-ranking, two-sided re-ranking, elasticity-inspired re-ranking, independence-based regularization, and error-parity regularization—we can conduct a controlled benchmarking study that bridges the methodological gap between post-hoc and in-processing approaches. The study deliberately limits the base recommenders to BPR-MF, representing a classic static, pairwise collaborative-filtering paradigm, and SASRec, representing a self-attentive sequential architecture. This choice does not aim to showcase a diverse model portfolio; rather, it isolates two fundamentally distinct learning strategies so that the impact of each fairness intervention can be examined under a unified experimental protocol. By doing so, the thesis provides the first systematic comparison of accuracy–fairness frontiers across paradigms while explicitly addressing the under-explored problem of head-vs-tail group fairness in recommendation.

# Chapter 3

# Methodology

## 3.1 Experimental Design

This thesis is designed as a controlled reproducibility study that compares multiple fairness-aware methods under a unified recommendation pipeline. Figure 3.1 shows an overview of our approach. The central idea is to apply different fairness methods to the same base recommenders, on the same datasets, with the same group definitions and evaluation metrics, so that any differences in outcomes can be attributed to the fairness mechanisms themselves rather than to confounding factors. We focus on item-side group fairness defined by item popularity. For each dataset, items are partitioned into head and tail groups based on their interaction frequency in the training data (e.g., top x% most popular items vs. the remaining long tail). Fairness is then quantified in terms of how exposure is distributed between these groups, while utility is measured by standard ranking metrics such as Recall@k and NDCG@k. Within this setting, we consider two base recommender models that represent distinct modeling paradigms: a static collaborative filtering model trained with Bayesian Personalized Ranking (BPR-MF), and a self-attentive sequential model (SASRec). On top of these base models, we apply a small but diverse set of fairness-aware methods that intervene at different points in the pipeline: in-processing regularization (independence-based and error-parity based) and post-hoc re-ranking (max-min provider-style re-ranking, two-sided re-ranking, and elasticity-based re-ranking). For each method, we sweep its fairness trade-off hyperparameter(s) to obtain an accuracy–fairness frontier.

To answer these questions mentioned in chapter 1, all experiments share the same high-level protocol: (i) fix datasets, data splits, and head–tail group definitions; (ii) train base models with a common set of hyperparameters; (iii) apply each fairness intervention with a grid of trade-off settings; and (iv) evaluate all resulting configurations with identical utility and fairness metrics. The remainder of this chapter details the datasets, base models, fairness methods, and evaluation procedures that implement this design.

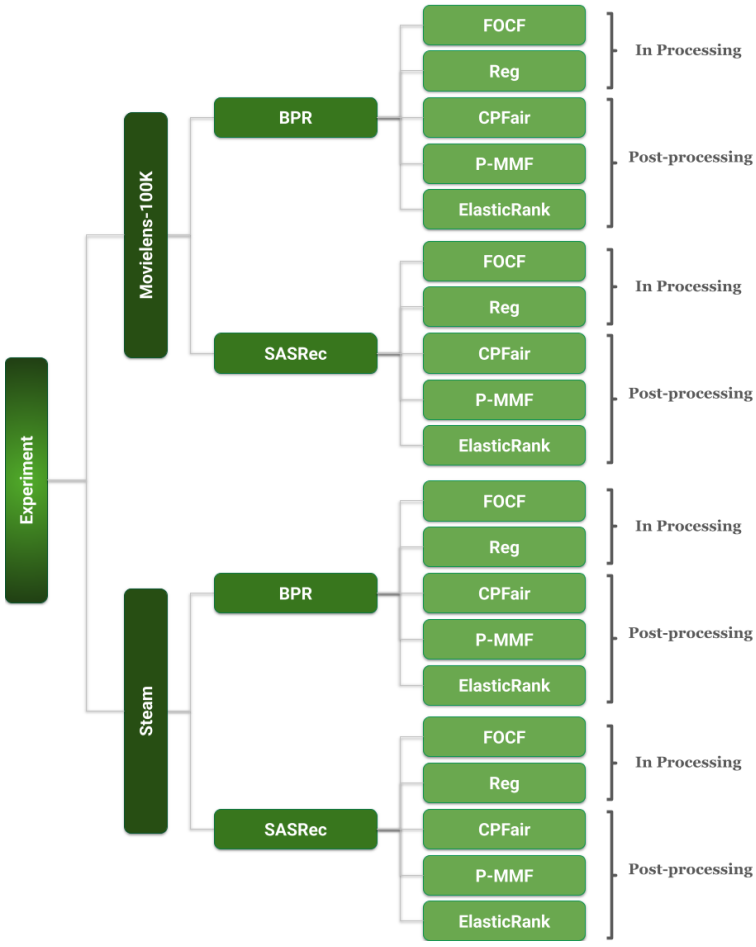



**Figure 3.1:** Overview of the experimental design architecture. The study systematically combines two datasets, two base recommendation models, and five fairness intervention methods (categorized as in-processing or post-processing).

### 3.1.1 Evaluation Protocol

To comprehensively analyze the performance of each fairness-aware model, we evaluate results across two primary dimensions: Utility (measured by Normalized Discounted Cumulative Gain, NDCG@K) and Fairness (measured by Tail Exposure, $E_{\text{Tail}}$)

Mapping the Fairness-Utility Frontier: For each baseline model (BPR-MF, SASRec) and each intervention method ($\mathcal{M}$), we sweep the fairness model's fairness hyperparameter ($\lambda$ or $t$) across a wide range of values. Each value yields a specific (NDCG@K, $E_{\text{Tail}}$) pair, allowing us to map the Fairness-Utility Frontier (Pareto Frontier) for that fairness model. This frontier visualizes the spectrum of achievable performance, ranging from maximum accuracy (minimal fairness) to maximum fairness (minimal accuracy).

Defining the Operating Point: To ensure a fair and consistent comparison between fairness models, we select a standardized operating point on each generated frontier. The optimal performance is defined as the point that yields the highest absolute Tail Exposure ($E_{\text{Tail}}$) while accepting a drop in NDCG@K of no more than 1% relative to the model's maximum achievable NDCG@K.

This consistent evaluation procedure ensures that all reported comparisons measure the efficacy of each technique in achieving actionable fairness gains while maintaining platform utility within an industrially tolerable loss threshold. The numerical results of this trade-off analysis form the core of our findings presented in Chapter 4.

## 3.2 Fairness definition

The focus of this study is entirely on mitigating popularity bias inherent in historical interaction data by manipulating the exposure disparity between item groups (head and tail). We acknowledge the fundamental accuracy-fairness trade-off inherent in recommender systems, where improving one metric typically degrades the other. Therefore, our objective is not to enforce a predetermined equitable outcome, but rather to systematically measure the maximum achievable fairness benefit for a minimal loss in recommendation accuracy (utility). The precise methodology for quantifying this trade-off is established in Section 3.1.1.

Our grouping strategy uses the Pareto Principle (80/20 rule) [16]. The Pareto Principle observes that "roughly 80% of the effects come from 20% of the causes". The 80%

threshold is widely adopted in fairness-aware recommendation work. With that being said, We partition the items into two mutually exclusive groups:

- **Head group ($\mathbb{I}_H$)** – the smallest subset of items that together account for 80 % of all observed user-item interactions in the dataset.
- **Tail group ($\mathbb{I}_T$)** – the remaining items, i.e., $\mathbb{I}_T = \mathbb{I} \setminus \mathbb{I}_H$, which contribute the final 20 % of interactions.

This definition is grounded in the Pareto principle and ensures that our group division reflects the true underlying bias severity of the system which has been addressed further in section 3.3.

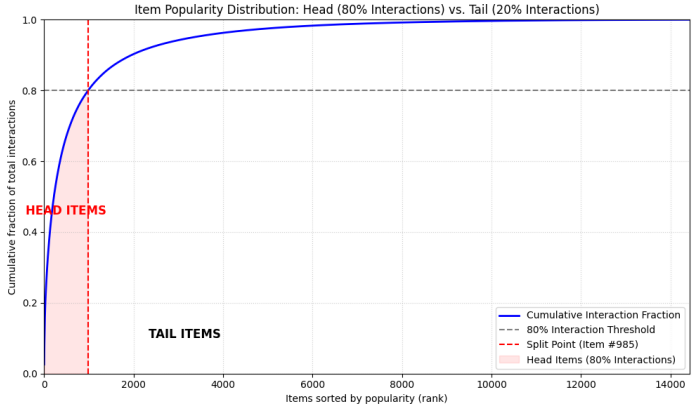
## 3.3 Datasets

To systematically benchmark the effectiveness and generalizability of fairness models, we selected two publicly available datasets that exhibit distinct consumption characteristics and varying degrees of inherent data concentration: MovieLens-100k and Steam. The choice of these two datasets is motivated by their differences across three critical dimensions relevant to recommender system evaluation and fairness modeling:

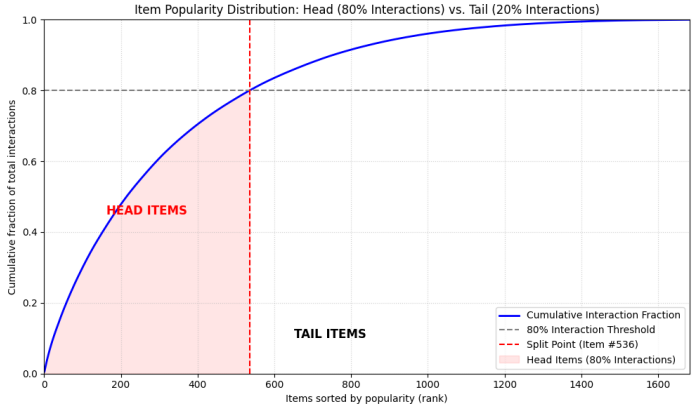
1. **Domain and Interaction Type:** We contrast collaborative filtering on traditional media ratings (MovieLens) against consumption patterns within the digital distribution platform domain (Steam), which involves game purchases and playtime logs. This ensures our findings are not confined to a single type of user behavior.
2. **Popularity Bias Severity (Head-vs-Tail Concentration):** The most critical difference is the variation in how popular items are within each set. Both datasets follow a "long-tail" pattern where a small number of items get most of the attention, but the exact severity of this popularity gap differs.

The core task of this study is to manage the trade-off between user utility (accuracy) and item exposure for items residing in the long tail. To quantify the difference in task difficulty and the inherent severity of data bias between our chosen datasets, we analyzed the Cumulative Distribution Function (CDF) of item interactions. We define the Head group as the set of most popular items that collectively account for 80% of all recorded user interactions. The remaining items constitute the Tail group, which holds

the sparse final 20% of interactions. The proportion of the total item catalog required to satisfy the 80% interaction threshold serves as a direct, quantifiable indicator of popularity bias concentration (Figures 3.2).




**(a)** Item Popularity Distribution for the Steam Dataset.




**(b)** Item Popularity Distribution for the MovieLens-100k Dataset.

**Figure 3.2:** Comparison of Item Popularity Distributions across datasets. Figure (a) illustrates the highly skewed curve for the Steam dataset, where the Head (80% of interactions) comprises only 7.0% of the item catalog. Figure (b) shows the MovieLens-100k distribution, which has a lower concentration of popularity bias, requiring 32% of the catalog for the same 80% of interactions.

The results in Figure 3.2 demonstrate a quantifiable difference in the severity of popularity bias: Steam exhibits an extremely high concentration of popularity bias. Only

7.0% of the total item catalog accounts for 80% of all observed activity. This presents a challenging environment for fairness interventions, as supporting the vast tail requires compensating for extremely low historical exposure. MovieLens-100k exhibits a moderate concentration of popularity bias. A significantly larger portion of the catalog, 32%, is needed to reach the 80% threshold. This environment is less skewed, suggesting that maintaining accuracy while achieving parity may be more tractable. By selecting datasets that operate in these two distinct regimes of bias severity, our study is positioned to evaluate the effectiveness of in-processing and post-proccessing techniques. This design ensures that the derived fairness-utility frontiers reflect the challenges of the diverse data distribution in the real-world.

## 3.4 Base Recommender Models

### 3.4.1 Bayesian Personalized Ranking

We use BPR as a static collaborative filtering model for this study, relying on a pairwise learning and implicit feedback data. The core mechanism involves learning a low-dimensional latent vector, $p_u$, for every user $u$, and an item vector, $q_i$, for every item $i$, where the preference score $\hat{x}ui$ is computed as their inner product. During training, the model processes triplets $(u, i, j)$, where $i$ is an observed (positive) item and $j$ is a sampled non-observed (negative) item. The model is optimized using the BPR-Opt criterion to ensure that the score for the positive item $\hat{x}ui$ is ranked higher than the score for the negative item $\hat{x}uj$. In the evaluation phase, the model produces a personalized ranking of all candidate items for user $u$ by sorting them in descending order of their predicted scores $\hat{x}ui$ [29].

### 3.4.2 Self-Attentive Sequential Recommendation

SASRec is the second baseline model, representing modern sequential recommendation, which focuses on capturing short-term user dynamics. It uses a Transformer-style architecture composed purely of self-attention layers, avoiding recurrent or convolutional modules. When predicting the next action, the model takes a user's recent interaction sequence $(i_1, i_2, \ldots, i_T)$ as input and employs self-attention to adaptively assign weights to previous items, generating a contextual user representation $s_u$. This representation $s_u$ is then used to compute a preference score $\hat{x}_{uj}$ for a candidate item $j$, which determines the item's position in the final ranked list. This approach allows the model to selectively focus on the most relevant prior actions, balancing the long-range

dependencies of RNNs with the efficiency of Markov Chains [18].

## 3.5 Fairness Models

### 3.5.1 Independence regularizer(Reg)

The aim of this regularizer is to make the predicted relevance scores statistically independent of the head-vs-tail popularity flag. After each training minibatch the algorithm computes the average score of all items marked as "head" and the average score of all items marked as "tail". The squared difference between the two averages is multiplied by a tunable weight $\lambda$ and added to the ordinary loss (BPR-MF or SASRec). The optimizer therefore minimizes a combined objective that simultaneously pushes positive items above their sampled negatives and reduces any systematic bias toward popular items [17].

### 3.5.2 FOCF

This regularizer targets the gap in prediction error between head and tail items. For every minibatch the method first computes the mean squared error for the head items and separately for the tail items. The absolute difference of these two means, scaled by a regularization coefficient $\lambda$, is added to the base loss. By penalising a large error gap the model is encouraged to treat popular and long-tail items with comparable accuracy [41].

### 3.5.3 Max-Min Provider-Style Re-ranking (P-MMF)

The max-min provider re-ranking method operates on the ranked list output by a base recommender and rescales item scores so that the *minimum* exposure received by any protected group (the Tail) is maximized while the *maximum* exposure of the advantaged group (the Head) is kept as low as possible[36].

### 3.5.4 CPFair

The CPFair method, as a post-processing reranking method, simultaneously enforces fairness for both the provider side (item groups) and the consumer side (user groups) [25]. In this study, in the context of item-side popularity fairness, the consumer-side component is omitted. A single fairness constraint is applied to ensure a balance exposure across Head/Tail groups while preserving per-user relevance.

### 3.5.5 ElasticRank

Elasticity-based re-ranking [40] interprets the trade-off between utility and fairness as an elasticity curve. For every user $u$ the algorithm computes an *elasticity factor* $\eta_u$ that quantifies how much the user's utility (e.g., NDCG) changes when tail exposure is increased. Items are then re-weighted according to this factor, giving larger boosts to tail items for users with higher elasticity and that their utility can tolerate more fairness adjustment.

# Chapter 4

# Experimental Setup and Results

## 4.1 Experimental Setup

### 4.1.1 Dataset Preprocessing

In this study, we use MovieLens-100k and Steam. These two dataset are selected to represent different domain characteristics and sparsity levels, which is crucial for assessing the robustness of fairness trade-offs (RQ2). MovieLens-100k (ML-100k) dataset is a classic movie-rating benchmark, and Steam dataset, is a large-scale video-game platform dataset containing play-count information. The data preprocessing consisted of three sequential phases:

1. **Positive-feedback Definition.** We establish criteria for positive interaction signals: for *MovieLens-100k*, ratings $\geq 4$ are considered positive feedback; for the *Steam* dataset, a play count of at least 10 is regarded as a positive interaction.

2. **User- and Item-Value Filtering.** To mitigate the influence of extreme sparsity and noisy interactions, standard 5-core filtering was applied, where both users and items appearing in fewer than 5 positive interactions were removed. The resulting data forms the final interaction matrix $\mathcal{S}^{\text{train}}$ used consistently

throughout the study.

3. **Head–Tail Construction (Pareto Principle).** Our grouping strategy uses the Pareto Principle (or 80/20 rule) [16]. The Pareto Principle observes that "roughly 80% of the effects come from 20% of the causes". Item popularity $M_i$ (total positive interactions) guides our fairness grouping. Items are sorted by decreasing $M_i$. The *Head* group $\mathcal{G}$Head is defined as the smallest set of items accounting for 80% of all cumulative interactions ($\mathcal{S}^{\text{train}}$). Conversely, the remaining items constitute the *Tail* group $\mathcal{G}$Tail, which collectively contributed the final 20% of interactions.

**Table 4.1:** Post-filtering characteristics of the two benchmark datasets.

| Metric | MovieLens-100k | Steam |
|---|---|---|
| Users $\lvert\mathcal{U}\rvert$ | 943 | 26,167 |
| Items $\lvert\mathcal{I}\rvert$ | 1,349 | 8,475 |
| Positive Interactions $\lvert\mathcal{S}\rvert$ | 99,287 | 340,037 |
| Sparsity $= 1 - \frac{\lvert\mathcal{S}\rvert}{\lvert\mathcal{U}\rvert \cdot \lvert\mathcal{I}\rvert}$ | 0.9220 | 0.9985 |
| Head items $\lvert\mathcal{G}_{\text{Head}}\rvert$ | 525 (38.92% of items) | 872 (10.29% of items) |
| Tail items $\lvert\mathcal{G}_{\text{Tail}}\rvert$ | 824 (61.08% of items) | 7,603 (89.71% of items) |

Table 4.1 shows the statistics of our selected datasets. The Steam dataset is extremely sparse, with a sparsity level of 0.9985, meaning that only a very small fraction of all possible user–item pairs are observed as interactions. In contrast, ML-100k is less sparse, with sparsity 0.9220. This difference implies that models on Steam must learn from comparatively fewer observations per user and per item than on ML-100k. The popularity distribution also differs clearly between the two domains. In Steam, only 872 items (about 10.29% of the catalog) form the head group that accounts for 80% of all interactions, while the remaining 7,603 items (89.71%) form the tail. In ML-100k, the head group is larger in relative terms: 525 items (38.92%) cover 80% of interactions and 824 items (61.08%) are in the tail. Thus, popularity is more strongly concentrated on a small subset of items in Steam than in ML-100k.

### 4.1.2 Models implementations and parameters Setup

All experiments were conducted on the LIACS Vibranum Compute Cluster at Leiden University, utilizing a dedicated node equipped with an NVIDIA GeForce RTX 3090 GPU (24 GB VRAM) and a multi-core CPU. This setup ensured consistent computa-

tional resources for fair comparisons across all models.

We implemented our framework using the Python ecosystem, primarily leveraging PyTorch for model execution. The base recommenders (BPR-MF and SASRec) were implemented using standard configurations consistent with established literature. All fairness models, including Reg[1], FOCF [2], P-MMF [3], CPFair [4], and ElasticRank [5] integrated directly by utilizing the official implementations provided by the authors, ensuring that performance evaluation reflects the intended mechanisms of each proposed method.

Table 4.2 shows the hyperparameters used for base recommender models. Table 4.3 summarizes the fair models parameters and sweep range. The sweep range used in this study is adopted by the suggested range in the original papers. To ensure reproducibility, all experiments were initialized using a fixed random seed of 42.

**Table 4.2:** Hyper-parameter settings for the base recommender models

| Model | Dataset | Parameter | Value |
|---|---|---|---|
| BPR | MI-100k | Learning rate | $10 \times 10^{-3}$ |
| | | epoch | 100 |
| | | Embedding size | 64 |
| | | early stopping patience | 5 |
| BPR | Steam | Learning rate | $10 \times 10^{-3}$ |
| | | epoch | 100 |
| | | Embedding size | 64 |
| | | early stopping patience | 5 |
| SASRec | MI-100k | Learning rate | $10 \times 10^{-3}$ |
| | | epochs | 100 |
| | | early stopping patience | 5 |
| | | dropout | 0.5 |
| | Steam | Learning rate | $10 \times 10^{-3}$ |
| | | epochs | 100 |
| | | early stopping patience | 5 |
| | | dropout | 0.5 |

[1]https://github.com/XuChen0427/FairDiverse/tree/master?tab=readme-ov-file
[2]https://github.com/TangJiakai/RecBole-FairRec
[3]https://github.com/XuChen0427/P-MMF
[4]https://github.com/rahmanidashti/CPFairRecSys
[5]https://github.com/XuChen0427/ElasticRank

| Model | Parameter | Sweep Range |
|---|---|---|
| Reg | Lambda | [0,8] |
| FOCF | Lambda | [0,5] |
| P-MMF | Gamma | [0,2] |
| | Lambda | [0,1] |
| CPFair | Lambda | [0,1] |
| ElasticRank | Anchor | [0.5, 0.95] |
| | Lambda | [0,2] |

**Table 4.3:** Fair Model parameters and sweep ranges

## 4.2 Evaluation Metrics

We evaluate all methods along two dimensions: (i) user-side accuracy, and (ii) item-side fairness based on exposure to popularity-based groups (head vs. tail). All metrics are computed at cutoff $K \in \{5, 10, 20, 50\}$ unless stated otherwise.

### 4.2.1 User-accuracy Metrics

Let $\mathcal{U}$ be the set of test users. For each user $u \in \mathcal{U}$, let $R_u^K = (i_1, \ldots, i_K)$ denote the top-$K$ recommendation list, and $G_u$ the set of relevant items for $u$ in the test set.

#### Recall@$K$

Recall@$K$ measures the fraction of relevant items that appear in the top-$K$ list:

$$\text{Recall@}K(u) = \frac{|R_u^K \cap G_u|}{|G_u|}. \tag{4.1}$$

We report the average over all test users:

$$\text{Recall@}K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \text{Recall@}K(u). \tag{4.2}$$

#### NDCG@$K$

We consider binary relevance, i.e., $\text{rel}_{u,i} = 1$ if $i \in G_u$ and 0 otherwise. The Discounted Cumulative Gain at cutoff $K$ is

$$\text{DCG@}K(u) = \sum_{j=1}^{K} \frac{\text{rel}_{u,i_j}}{\log_2(j+1)}, \tag{4.3}$$

where $i_j$ is the item at position $j$ in $R_u^K$. The corresponding ideal DCG is obtained by placing all relevant items at the top:

$$\text{IDCG@}K(u) = \sum_{j=1}^{\min(K,|G_u|)} \frac{1}{\log_2(j+1)}. \tag{4.4}$$

The normalized DCG is then

$$\text{NDCG@}K(u) = \frac{\text{DCG@}K(u)}{\text{IDCG@}K(u)}, \tag{4.5}$$

and we report the user average:

$$\text{NDCG@}K = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \text{NDCG@}K(u). \tag{4.6}$$

### 4.2.2 Fairness Metrics

To study item-side fairness, we focus on how exposure is distributed across popularity-based item groups. Let $\mathcal{G} = \{\text{head}, \text{tail}\}$ denote the two groups. For a fixed cutoff $K$, we define exposure in terms of how often items from each group appear in the top-$K$ recommendations.

#### Group Exposure

The (unnormalized) exposure of group $g \in \mathcal{G}$ is

$$E_g = \frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \sum_{j=1}^{K} \mathbb{I}\{i_{u,j} \in g\}, \tag{4.7}$$

where $\mathbb{I}\{\cdot\}$ is the indicator function and $i_{u,j}$ denotes the item at position $j$ in user $u$'s top-$K$ list. Thus, $E_g$ can be interpreted as the average number of top-$K$ recommendation slots occupied by group $g$ per user.

We also define normalized exposure shares

$$p_g = \frac{E_g}{\sum_{g' \in \mathcal{G}} E_{g'}}. \tag{4.8}$$

These group-level exposures $E_g$ (and their normalized shares $p_g$) serve as the basis for

our fairness metrics, including MMF@K, EF@K, MinMaxRatio@K, Entropy@K, and the Gini coefficient of exposure.

### Max–min Exposure Fairness (MMF)

To quantify how well the worst-off popularity group is supported, we use a max–min exposure fairness index inspired by P–MMF [36]. Let $\mathcal{G} = \{\text{head}, \text{tail}\}$ denote the two popularity-based item groups, and let $E_g$ be the cumulative exposure of group $g$ at cutoff $K$ across all users (as defined in the previous subsection). We then define

$$\text{MMF@}K = \frac{\min\left(E_{\text{head}}, E_{\text{tail}}\right)}{E_{\text{head}} + E_{\text{tail}}}. \tag{4.9}$$

Intuitively, MMF@$K$ measures the fraction of total exposure allocated to the worse-off group. Higher values indicate that even the least-exposed group which is tail group in our study, receives a larger share of the overall exposure, whereas low values indicate that one group is strongly under-exposed.

### Elastic Fairness (EF)

Following Xu et al. [40], we adopt the Elastic Fairness (EF) metric, which summarizes fairness performance across a continuum of elasticity parameters. In our setting, fairness is defined in terms of exposure. We therefore instantiate the group utility vector $\mathbf{v} = (v_g)_{g\in\mathcal{G}}$ using the group exposures $E_g$ defined above:

$$v_g := E_g, \tag{4.10}$$

We first normalize group utilities to obtain a distribution over groups:

$$\bar{v}_g = \frac{v_g}{\sum_{g'\in\mathcal{G}} v_{g'}}. \tag{4.11}$$

For a given tax base (elasticity parameter) $t$, the corresponding fairness score is

$$f(\mathbf{v}; t) = \text{sign}(1-t)\left(\sum_{g\in\mathcal{G}} \bar{v}_g^{\,1-t}\right)^{1/t}. \tag{4.12}$$

Different values of $t$ recover different classical notions of fairness and place varying

emphasis on low-utility ("poor") groups.

The Elastic Fairness metric EF@K is then defined as the normalized area under the EF-curve:

$$\mathrm{EF@}K = \frac{1}{Z} \int_{1-M}^{1+M} f(\mathbf{v}; t)\, \mathrm{d}t, \tag{4.13}$$

where $M \geq 0$ controls the integration range around $t = 1$ and $Z$ is a normalization constant (e.g., $Z = 2M\,|\mathcal{G}|$). In this formulation, higher EF@K indicates better exposure fairness on average across the range of elasticity parameters.

### Min–max Ratio

The min–max ratio is a simple dispersion measure over group exposures:

$$\mathrm{MinMaxRatio@}K = \frac{\min_{g \in \mathcal{G}} E_g}{\max_{g \in \mathcal{G}} E_g}. \tag{4.14}$$

For two groups (head and tail), this reduces to

$$\mathrm{MinMaxRatio@}K = \frac{\min(E_{\mathrm{head}}, E_{\mathrm{tail}})}{\max(E_{\mathrm{head}}, E_{\mathrm{tail}})}. \tag{4.15}$$

Values close to 1 indicate that both groups receive similar exposure, whereas values close to 0 indicate strong imbalance.

### Exposure Entropy

We also report the Shannon entropy of the group exposure distribution:

$$\mathrm{Entropy@}K = -\sum_{g \in \mathcal{G}} p_g \log p_g. \tag{4.16}$$

For two groups, the maximum entropy is $\log 2$, achieved when exposure is perfectly balanced between head and tail. Lower entropy indicates more concentrated exposure.

### Gini Coefficient of Exposure

Finally, we measure exposure inequality at the item level using the Gini coefficient. Let $x_1, \ldots, x_n$ denote the per-item exposures, sorted in non-decreasing order, and let

$$\mu = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{4.17}$$

be the mean exposure. The Gini coefficient is defined as

$$\text{Gini@}K = \frac{1}{n\,\mu} \sum_{i=1}^{n} (2i - n - 1)\, x_i. \tag{4.18}$$

A value of Gini@$K = 0$ corresponds to perfectly equal exposure across items, while values closer to 1 indicate that exposure is concentrated on a small subset of items.

## 4.3 Experimental Results

In this section we evaluate the trade-off between recommendation utility and tail-item exposure for each base recommender and dataset under a fixed utility budget of RelativeNDCG $\geq 0.990$ (no more than a 1% loss relative to the baseline). Within this admissible region we maximise RelativeMMF (the tail-exposure metric), which has been described in section 3.1.1.

As been seen earlier in section 4.1.1 and Table 4.1, the biases within the two selected dataset, present notable contrasts. The MovieLens-100k dataset is characterized by a relatively balanced distribution, containing 38.92% head items and 61.08% tail items. In sharp contrast, the Steam dataset exhibits a pronounced long-tail distribution, with only 10.29% head items and 89.71% tail items. This fundamental difference in intrinsic item popularity bias, serves as the principal driver of the divergent fairness-utility trade-offs observed across our experiments.

### 4.3.1 Detailed Performance of Fairness Methods

#### Performance of BPR on the MovieLens-100k Dataset

Table 4.4 demonstrates the evaluation of the fairness models implemented on BPR model and the MovieLens-100k dataset. The operating points are selected by focusing on maximizing Relative MMF, a metric representing tail exposure gain, and adhering strictly to the Relative NDCG budget of 0.990. At the initial cutoff of $K = 5$, where the Baseline MMF stands at 0.286, the post-processing methods demonstrated clear superiority, with CPF achieving the highest Relative MMF of 1.706. Critically, this highest Relative MMF score coincided with the optimal performance across all secondary fairness metrics (Elastic Fairness, Entropy and Gini ), resulting an effective fairness method. Moving to $K = 10$, the baseline MMF remained stable at 0.282. Here, the CPF model failed to meet the strict NDCG budget, leaving ER (Elasti-

cRank) as the top performer among admissible methods, achieving a Relative MMF of 1.753. This method, along with the admissible P-MMF, achieved near-perfect scores for the other fairness metrics (Entropy $\approx 1.000$, Gini $\approx 0.01$), confirming that the approach that yields the highest Relative MMF simultaneously drives the highest levels of parity and distribution improvement. This trend persisted at $K = 20$, where the baseline MMF was 0.288. With CPF again falling below the acceptable utility threshold, ER once more emerged with the highest Relative MMF (1.688), closely followed by P-MMF (1.678). Both post-processing models continued to dominate the secondary fairness metrics, validating the consistent coupling between maximized tail exposure gain and optimal list distribution quality. Finally, at the larger cutoff of $K = 50$, where the baseline MMF rose to 0.325, CPF was the top performer with a Relative MMF of 1.509, again leading the field in achieving the best secondary fairness scores. Across all tested cutoffs, post-processing techniques consistently achieved a markedly better trade-off by producing Relative MMF gains ranging from 1.509 to 1.753. In contrast, in-processing methods delivered marginal gains, demonstrating that for the balanced MovieLens dataset, post-processing methods achieved a far superior fairness-utility trade-off.

These results are consistent with the nature of the MovieLens-100k data set, whose head–tail distribution is relatively mild (the proportion of head items is far from extreme). Consequently, post-processing approaches can substantially boost tail exposure without jeopardizing accuracy. In contrast, the in-processing techniques (FOCF and Reg) achieve only modest RelativeMMF improvements (the best being 1.06) but preserve higher utility; for example, FOCF at K=5 reaches a RelativeNDCG of 1.096, showing that these methods prioritize accuracy over fairness. Overall, for this dataset and base model combination, the post-processing methods provide a far superior fairness-utility trade-off, while the in-processing methods offer a utility-focused alternative with limited fairness gains.

**Performance of BPR on the Steam Dataset**

Table 4.5 demonsterates the evaluation of all selected fair models, implementd on BPR base model and Steam dataset. The results reveals a distinct pattern compared to MovieLens-100k, primarily due to the Steam data exhibiting a much larger disparity between head and tail items, as evidenced by the significantly lower Baseline MMF across all cutoffs (ranging from 0.113 at $K = 5$ to 0.346 at $K = 50$). The evaluation adheres to the constraint of Relative NDCG $\geq 0.990$.

**Table 4.4:** Performance Comparison and Trade-off Analysis for BPR on the *MovieLens-100k* Dataset. Results show performance for selected operating points across four cutoff lengths ($K = 5, 10, 20, 50$). Each configuration is optimized to maximize Relative MMF (Tail Exposure) subject to a utility budget constraint of **Relative NDCG** $\geq 0.990$ (max 1% loss relative to the baseline).

| Method | Paradigm | $K$ | Accuracy | | | Fairness | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | NDCG | RelNDCG | Recall | MMF | EF | Entropy | Gini | RelMMF |
| Baseline | — | 5 | 0.144 | – | 0.037 | 0.286 | -0.442 | 0.863 | 0.215 | – |
| CPF | Post | 5 | 0.150 | 1.042 | 0.041 | 0.487 | -0.007 | 1.000 | 0.013 | 1.706 |
| P-MMF | Post | 5 | 0.147 | 1.018 | 0.036 | 0.482 | -0.013 | 0.999 | 0.018 | 1.689 |
| ER | Post | 5 | 0.148 | 1.027 | 0.040 | 0.486 | -0.008 | 0.999 | 0.013 | 1.702 |
| FOCF | In | 5 | 0.158 | 1.096 | 0.053 | 0.289 | -0.432 | 0.867 | 0.211 | 1.011 |
| Reg | In | 5 | 0.148 | 1.028 | 0.038 | 0.290 | -0.427 | 0.869 | 0.210 | 1.017 |
| Baseline | — | 10 | 0.142 | – | 0.049 | 0.282 | -0.453 | 0.859 | 0.218 | – |
| CPF | Post | 10 | 0.135 | 0.951 | 0.037 | 0.490 | -0.005 | 1.000 | 0.011 | 1.734 |
| P-MMF | Post | 10 | 0.141 | 0.992 | 0.048 | 0.494 | -0.002 | 1.000 | 0.006 | 1.751 |
| ER | Post | 10 | 0.142 | 1.000 | 0.051 | 0.495 | -0.002 | 1.000 | 0.010 | 1.753 |
| FOCF | In | 10 | 0.148 | 1.036 | 0.045 | 0.290 | -0.430 | 0.868 | 0.211 | 1.026 |
| Reg | In | 10 | 0.148 | 1.041 | 0.061 | 0.284 | -0.447 | 0.861 | 0.216 | 1.006 |
| Baseline | — | 20 | 0.145 | – | 0.081 | 0.288 | -0.4347 | 0.866 | 0.212 | – |
| CPF | Post | 20 | 0.143 | 0.985 | 0.074 | 0.485 | -0.010 | 0.999 | 0.015 | 1.684 |
| P-MMF | Post | 20 | 0.146 | 1.010 | 0.079 | 0.483 | -0.012 | 0.999 | 0.017 | 1.678 |
| ER | Post | 20 | 0.144 | 0.995 | 0.080 | 0.486 | -0.009 | 0.999 | 0.014 | 1.688 |
| FOCF | In | 20 | 0.145 | 1.001 | 0.073 | 0.304 | -0.386 | 0.887 | 0.196 | 1.057 |
| Reg | In | 20 | 0.149 | 1.032 | 0.093 | 0.302 | -0.393 | 0.884 | 0.198 | 1.049 |
| Baseline | — | 50 | 0.172 | – | 0.181 | 0.325 | -0.330 | 0.910 | 0.175 | – |
| CPF | Post | 50 | 0.173 | 1.004 | 0.179 | 0.490 | -0.004 | 1.000 | 0.010 | 1.509 |
| P-MMF | Post | 50 | 0.172 | 1.001 | 0.181 | 0.330 | -0.316 | 0.915 | 0.170 | 1.016 |
| ER | Post | 50 | 0.172 | 0.999 | 0.180 | 0.482 | -0.013 | 0.999 | 0.180 | 1.483 |
| FOCF | In | 50 | 0.175 | 1.020 | 0.179 | 0.331 | -0.314 | 0.916 | 0.169 | 1.019 |
| Reg | In | 50 | 0.180 | 1.048 | 0.186 | 0.335 | -0.303 | 0.920 | 0.165 | 1.032 |

At $K = 5$ the post-processing technique P-MMF attains the largest increase in RelativeMMF of 2.89 and also yields the most favorable secondary fairness scores (the lowest Gini, the highest Entropy, and the lowest EF). However, this improvement comes at a steep cost to recommendation quality, as its RelativeNDCG drops to 0.75, which exceeds the allowed 1% loss (the required threshold is RelativeNDCG $\geq 0.990$). Consequently, the best admissible solution is the in-processing method Reg, which still raises RelativeMMF to 1.18 while preserving a RelativeNDCG of 1.01, comfortably within the utility budget.

For $K = 10$ the situation is similar. P-MMF again delivers the strongest fairness gain at RelativeMMF of 2.75, yet its RelativeNDCG falls to 0.75, far below the 0.99 requirement. The in-processing methods Reg and FOCF remains the only approaches that satisfies the accuracy constraint, offering a RelativeMMF of 1.23 and 1.139, and a RelativeNDCG of 0.991 and 1.017 respectively

At $K = 20$ the baseline MMF has risen modestly to 0.140, and P-MMF still provides the highest RelativeMMF of 2.20 together with the best Gini, Entropy and EF values, but its RelativeNDCG is only 0.78. Again, this level of accuracy loss is unacceptable. Among the methods that meet the 1% NDCG budget, the in-processing technique Reg yields the highest RelativeMMF at 1.09 while keeping RelativeNDCG at 1.008, and its secondary fairness metrics are only slightly better than the baseline.

When the recommendation list is extended to $K = 50$, the baseline MMF has increased to 0.346, and all methods stay within the accuracy requirement (RelativeNDCG $\geq$ 0.990). Here the in-processing approaches FOCF and Reg achieve the highest RelativeMMF of 1.022, marginally surpassing the post-processing methods. Nonetheless, among the post-processing models, ElasticRank achieved the highest RelativeMMF at 1.015, and RelativeNDCG of 0.991, adhering the utility budget.

n summary, for the Steam data set the in-processing methods (FOCF, Reg) provide the most reliable trade-off when a strict utility constraint (no more than a 1% NDCG loss) must be respected, as they consistently satisfy the RelativeNDCG threshold and achieve the greatest admissible RelativeMMF. The post-processing methods (CPF, P-MMF, ER) are capable of delivering considerably larger fairness gains and superior secondary fairness diagnostics, but these gains are realized only at the expense of a substantial drop in NDCG for the smaller cutoffs, making them unsuitable under the defined budget. At the largest cutoff ($K = 50$), where the accuracy loss is modest, post-processing methods become competitive in fairness, yet the in-processing approaches still retain a slight edge in RelativeMMF while meeting the accuracy requirement.

.

**Performance of SASRec on the MovieLens-100k Dataset**

The SASRec model on the MovieLens-100k data set exhibits a relatively modest baseline tail-exposure (MMF) that rises steadily with the recommendation list length:

**Table 4.5:** Performance Comparison and Trade-off Analysis for BPR on the *Steam* Dataset. Results show performance for selected operating points across four cutoff lengths ($K = 5, 10, 20, 50$). Each configuration is optimized to maximize Relative MMF (Tail Exposure) subject to a utility budget constraint of **Relative NDCG** $\geq 0.990$ (max 1% loss relative to the baseline)

| Method | Paradigm | $K$ | Accuracy | | | Fairness | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | NDCG | RelNDCG | Recall | MMF | EF | Entropy | Gini | RelMMF |
| Baseline | — | 5 | 0.259 | – | 0.300 | 0.113 | -1.618 | 0.510 | 0.387 | – |
| CPF | Post | 5 | 0.237 | 0.917 | 0.287 | 0.182 | -0.908 | 0.685 | 0.318 | 1.607 |
| P-MMF | Post | 5 | 0.195 | 0.754 | 0.245 | 0.327 | -0.324 | 0.912 | 0.173 | 2.889 |
| ER | Post | 5 | 0.239 | 0.923 | 0.290 | 0.188 | -0.899 | 0.691 | 0.308 | 1.659 |
| FOCF | In | 5 | 0.265 | 1.022 | 0.298 | 0.130 | -1.381 | 0.558 | 0.370 | 1.148 |
| Reg | In | 5 | 0.262 | 1.010 | 0.306 | 0.134 | -1.338 | 0.567 | 0.366 | 1.179 |
| Baseline | — | 10 | 0.278 | – | 0.354 | 0.118 | -1.538 | 0.525 | 0.382 | – |
| CPF | Post | 10 | 0.258 | 0.927 | 0.344 | 0.164 | -1.038 | 0.645 | 0.336 | 1.387 |
| P-MMF | Post | 10 | 0.208 | 0.749 | 0.291 | 0.326 | -0.328 | 0.911 | 0.174 | 2.749 |
| ER | Post | 10 | 0.274 | 0.985 | 0.355 | 0.141 | -1.251 | 0.581 | 0.361 | 1.190 |
| FOCF | In | 10 | 0.282 | 1.017 | 0.351 | 0.135 | -1.322 | 0.571 | 0.365 | 1.139 |
| Reg | In | 10 | 0.275 | 0.991 | 0.360 | 0.146 | -1.207 | 0.599 | 0.354 | 1.230 |
| Baseline | — | 20 | 0.302 | – | 0.442 | 0.140 | -1.265 | 0.584 | 0.360 | – |
| CPF | Post | 20 | 0.284 | 0.938 | 0.439 | 0.164 | -1.040 | 0.644 | 0.336 | 1.172 |
| P-MMF | Post | 20 | 0.236 | 0.782 | 0.385 | 0.308 | -0.375 | 0.891 | 0.192 | 2.200 |
| ER | Post | 20 | 0.296 | 0.978 | 0.432 | 0.156 | -1.033 | 0.621 | 0.345 | 1.112 |
| FOCF | In | 20 | 0.308 | 1.020 | 0.445 | 0.151 | -1.155 | 0.612 | 0.349 | 1.078 |
| Reg | In | 20 | 0.305 | 1.008 | 0.449 | 0.153 | -1.141 | 0.616 | 0.348 | 1.089 |
| Baseline | — | 50 | 0.340 | – | 0.610 | 0.346 | -0.278 | 0.930 | 0.154 | – |
| CPF | Post | 50 | 0.321 | 0.946 | 0.608 | 0.350 | -0.268 | 0.934 | 0.150 | 1.012 |
| P-MMF | Post | 50 | 0.337 | 0.993 | 0.609 | 0.349 | -0.271 | 0.933 | 0.151 | 1.008 |
| ER | Post | 50 | 0.337 | 0.991 | 0.604 | 0.351 | -0.266 | 0.935 | 0.149 | 1.015 |
| FOCF | In | 50 | 0.343 | 1.008 | 0.604 | 0.353 | -0.260 | 0.937 | 0.147 | 1.022 |
| Reg | In | 50 | 0.343 | 1.011 | 0.606 | 0.353 | -0.260 | 0.937 | 0.147 | 1.022 |

0.203 at $K = 5$, 0.217 at $K = 10$, 0.239 at $K = 20$ and 0.281 at $K = 50$. Because every fairness-enhancing configuration was optimized under the same strict utility budget (Table 4.6).

At the smallest cutoff ($K = 5$), the in-processing approach Reg delivers the largest RelativeMMF (1.207) and simultaneously satisfies the utility requirement with a RelativeNDCG of 1.023. Its secondary fairness diagnostics are the most favorable among all methods: the Gini coefficient drops to 0.255, Entropy rises to 0.804 and the Elastic Fairness (EF) score is the least negative (-0.585). The other in-processing method

FOCF also meets the budget (RelativeNDCG = 0.994) but attains a lower RelativeMMF (1.135) and slightly weaker secondary metrics. All post-processing techniques (CPF, P-MMF, ER) respect the NDCG constraint as well but each provides a smaller RelativeMMF (ranging from 1.095 to 1.112) and modestly higher Gini values ($\approx$ 0.27–0.28), lower Entropy ($\approx$ 0.76–0.77) and more negative EF ($\approx$ -0.67 to -0.68). Thus, for $K = 5$ the method that maximizes tail exposure also delivers the best secondary fairness outcomes, and this method belongs to the in-processing models.

When the list length is increased to $K = 10$, the baseline MMF grows to 0.217. Here FOCF attains the highest RelativeMMF (1.153) while keeping RelativeNDCG at 1.019, comfortably above the required threshold. Its secondary metrics are again the most advantageous (Gini = 0.250, Entropy= 0.811, EF =-0.566). The sibling in-processing method Reg follows closely with a RelativeMMF of 1.138 and a slightly higher RelativeNDCG (1.030), and it likewise improves Gini (0.253), Entropy (0.806) and EF (-0.578). The post-processing models satisfy the NDCG budget but only barely (CPF at exactly 0.990, P-MMF at 0.999, ER at 0.992). Their RelativeMMF scores are noticeably lower (1.115 for CPF, 1.034 for P-MMF, 1.148 for ER) and their secondary fairness numbers are correspondingly weaker (higher Gini, lower Entropy, more negative EF). Consequently, the method that yields the greatest tail-exposure gain also provides the best overall fairness profile, and both of these top performers are in-processing.

At $K = 20$ the baseline MMF reaches 0.239. The in-processing technique Reg again leads with a RelativeMMF of 1.145 and a RelativeNDCG of 1.035. Its Gini (0.226) and Entropy (0.847) are the best among all candidates, and its EF (-0.480) is the least adverse. FOCF is a very close second (RelativeMMF = 1.105, RelativeNDCG = 1.029) and shows comparable secondary metrics. All post-processing approaches satisfy the NDCG constraint (CPF at 0.990, P-MMF at 0.998, ER at 0.994, CPF at 0.990) but achieve substantially lower RelativeMMF values (1.013–1.087) and exhibit higher Gini coefficients ($\approx$ 0.24–0.26) and lower Entropy ($\approx$ 0.79 – 0.80). Hence, the pattern observed at smaller cutoffs persists: the highest RelativeMMF coincides with the most favorable secondary fairness scores, and the winner belongs to the in-processing family.

For the longest list ($K = 50$) the baseline MMF is 0.281. Both in-processing methods Reg and FOCF surpass the 1% NDCG budget comfortably (RelativeNDCG =1.051 for Reg and 1.039 for FOCF) and achieve the highest RelativeMMF values (1.093 for Reg, 1.084 for FOCF). Their secondary fairness indicators are also superior: Reg records

the lowest Gini (0.193), the highest Entropy (0.890) and the least negative EF (-0.378). The post-processing models meet the accuracy requirement but lag behind in all fairness dimensions, with RelativeMMF ranging from 1.001 (P-MMF) to 1.046 (ER), Gini values around 0.21–0.22, Entropy near 0.86–0.87 and EF values between -0.416 and -0.444. Thus, even at the largest cutoff, the method that maximizes tail exposure also yields the best secondary fairness outcomes, and it is again an in-processing technique.

Across every cutoff length, the in-processing approaches, particularly Reg, with FOCF a very close competitor, dominate the trade-off landscape. They consistently satisfy the 1% NDCG budget, achieve the greatest RelativeMMF gains, and produce the most equitable secondary fairness profiles (lowest Gini, highest Entropy, least negative EF). Post-processing methods, while capable of modest fairness improvements, never reach the same level of tail-exposure enhancement and typically exhibit weaker secondary metrics, even when they just meet the accuracy constraint. Consequently, for SASRec on MovieLens-100k,in-processing methods, appear to be the most effective strategy for attaining a balanced accuracy–fairness trade-off.

**Table 4.6:** Performance Comparison and Trade-off Analysis for SASRec on the *MovieLens-100k* Dataset. Results show performance for selected operating points across four cutoff lengths ($K = 5, 10, 20, 50$). Each configuration is optimized to maximize Relative MMF (Tail Exposure) subject to a utility budget constraint of **Relative NDCG** $\geq 0.990$ (max 1% loss relative to the baseline)

| Method | Paradigm | $K$ | Accuracy | | | Fairness | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | NDCG | RelNDCG | Recall | MMF | EF | Entropy | Gini | RelMMF |
| Baseline | — | 5 | 0.341 | – | 0.126 | 0.203 | -0.780 | 0.728 | 0.297 | – |
| CPF | Post | 5 | 0.340 | 0.996 | 0.123 | 0.224 | -0.675 | 0.768 | 0.276 | 1.103 |
| P-MMF | Post | 5 | 0.338 | 0.992 | 0.125 | 0.223 | -0.682 | 0.765 | 0.277 | 1.095 |
| ER | Post | 5 | 0.339 | 0.996 | 0.125 | 0.226 | -0.667 | 0.770 | 0.270 | 1.112 |
| FOCF | In | 5 | 0.339 | 0.994 | 0.138 | 0.231 | -0.645 | 0.779 | 0.269 | 1.135 |
| Reg | In | 5 | 0.349 | 1.023 | 0.146 | 0.245 | -0.585 | 0.804 | 0.255 | 1.207 |
| Baseline | — | 10 | 0.339 | – | 0.199 | 0.217 | -0.709 | 0.755 | 0.283 | – |
| CPF | Post | 10 | 0.335 | 0.990 | 0.196 | 0.242 | -0.598 | 0.798 | 0.258 | 1.115 |
| P-MMF | Post | 10 | 0.338 | 0.999 | 0.198 | 0.224 | -0.675 | 0.768 | 0.276 | 1.034 |
| ER | Post | 10 | 0.336 | 0.992 | 0.196 | 0.249 | -0.568 | 0.804 | 0.253 | 1.148 |
| FOCF | In | 10 | 0.345 | 1.019 | 0.197 | 0.250 | -0.566 | 0.811 | 0.250 | 1.153 |
| Reg | In | 10 | 0.349 | 1.030 | 0.204 | 0.247 | -0.578 | 0.806 | 0.253 | 1.138 |
| Baseline | — | 20 | 0.344 | – | 0.287 | 0.239 | -0.609 | 0.794 | 0.261 | – |
| CPF | Post | 20 | 0.341 | 0.990 | 0.286 | 0.252 | -0.556 | 0.815 | 0.248 | 1.056 |
| P-MMF | Post | 20 | 0.344 | 0.998 | 0.286 | 0.242 | -0.596 | 0.799 | 0.258 | 1.013 |
| ER | Post | 20 | 0.342 | 0.994 | 0.287 | 0.260 | -0.528 | 0.825 | 0.241 | 1.087 |
| FOCF | In | 20 | 0.354 | 1.029 | 0.294 | 0.264 | -0.513 | 0.833 | 0.236 | 1.105 |
| Reg | In | 20 | 0.357 | 1.035 | 0.282 | 0.274 | -0.480 | 0.847 | 0.226 | 1.145 |
| Baseline | — | 50 | 0.371 | – | 0.449 | 0.281 | -0.457 | 0.857 | 0.219 | – |
| CPF | Post | 50 | 0.368 | 0.991 | 0.448 | 0.285 | -0.444 | 0.862 | 0.215 | 1.014 |
| P-MMF | Post | 50 | 0.372 | 1.002 | 0.449 | 0.281 | -0.456 | 0.857 | 0.219 | 1.001 |
| ER | Post | 50 | 0.370 | 0.997 | 0.448 | 0.294 | -0.416 | 0.868 | 0.206 | 1.046 |
| FOCF | In | 50 | 0.386 | 1.039 | 0.447 | 0.305 | -0.385 | 0.887 | 0.196 | 1.084 |
| Reg | In | 50 | 0.390 | 1.051 | 0.446 | 0.307 | -0.378 | 0.890 | 0.193 | 1.093 |

### Performance of SASRec on the Steam Dataset

The SASRec model on the Steam dataset exhibits extreme baseline tail-exposure values that rise sharply with the recommendation list length: 0.022 at $K = 5$, 0.035 at $K = 10$, 0.075 at $K = 20$, and 0.312 at $K! =!50$. The critical insight is that these baseline MMF values determine the available wiggle room for fairness interventions. At the smaller cutoffs where baseline tail-exposure is extremely low, fairness methods have substantial flexibility to boost fairness metrics without immediately violating the 1% NDCG loss ceiling. By contrast, at $K = 50$ where baseline MMF is already

0.312, the accuracy constraint becomes far more binding, severely limiting how much additional tail-exposure fairness methods can achieve (Table 4.7).

At the smallest cutoff ($K = 5$) the baseline MMF is just 0.022, a near-zero value that provides enormous room for fairness improvements. The post-processing method CPF exploits this freedom effectively, achieving the highest Relative MMF (1.900) among methods that respect the accuracy constraint (Relative NDCG = 0.993). Its secondary fairness metrics are notably strong: Gini drops substantially to 0.458, Entropy rises to 0.250, and EF improves sharply to $-4.635$. ER also performs exceptionally well at this cutoff, achieving a Relative MMF of 1.740 while maintaining utility constrain with Relative NDCG of 1.000 and delivering favorable secondary metrics at Gini = 0.443, Entropy = 0.231, EF = $-5.841$. The post-processing technique P-MMF also meets the NDCG budget (Relative NDCG = 1.000) and attains a Relative MMF of 1.384, though with slightly weaker secondary metrics (Gini = 0.470, Entropy = 0.196). By contrast, the in-processing methods struggle to capitalize on this available headroom. FOCF barely improves tail exposure (Relative MMF = 1.005) and leaves Gini and Entropy nearly unchanged from the baseline, suggesting that the direct regularization approach cannot exploit the flexibility afforded by the extremely low baseline. On the other hand, although Reg fails to meet the accuracy constraint just by 0.004, it achieves the highest RelMFF within in-processing methods with MMF of 0.03. Consequently, at $K = 5$ where baseline tail-exposure provides maximal flexibility, the post-processing methods, deliver substantially superior fairness gains while maintaining acceptable accuracy, whereas in-processing techniques appear to be struggling with extreme skewness and abundant optimization room.

At $K = 10$ the baseline MMF rises to 0.035, still extremely low and thus still providing substantial room for fairness gain. The post-processing method ER performs as the top performer, achieving the highest Relative MMF at 1.585 while maintaining strong Relative NDCG at 0.998. Its secondary fairness metrics, Gini = 0.439, Entropy = 0.311, EF = -3.042, substantially outperforming the baseline and delivering the best Gini score among all methods at this cutoff. Post-processing CPF follows closely ait Relative MMF at 1.562, Relative NDCG at 0.991 with similarly strong secondary metrics with Gini = 0.446, Entropy = 0.304, EF = -3.550. P-MMF remains compliant with Relative NDCG of 1.004, reaches a Relative MMF of 1.254, though with modestly weaker secondary metrics. Among the in-processing approaches, FOCF and Reg both meet the accuracy requirement with Relative NDCG = 0.998 and 0.991, respectively, but achieve markedly lower Relative MMF values 1.167 and 1.245 respectively. The

pattern established at $K = 5$ persists at $K = 10$: because the baseline tail-exposure remains negligibly small, post-processing techniques outperform in-processing methods in extracting tail-exposure gains while remaining well within the utility budget.

At $K = 20$ the baseline MMF increases to 0.075, still low in absolute terms but noticeably higher than at the preceding cutoffs. The available wiggle room for fairness improvements begins to contract, yet it remains substantial enough that post-processing methods continue to thrive. The post-processing method ER achieves the highest Relative MMF of 1.501 with Relative NDCG of 0.998. Its secondary metrics are outperforming other methods, with Gini = 0.387, Entropy = 0.507, EF = -5.841. Post-processing CPF follows closely with Relative MMF = 1.445, Relative NDCG = 0.991. P-MMF also meets the utility constrain at Relative NDCG = 1.002 but yields a substantially lower Relative MMF 1.175, and weaker secondary metrics. The in-processing methods FOCF and Reg meet the accuracy threshold but lag significantly in fairness gains with Relative MMF = 1.112 and 1.206 respectively. Once again, post-processing approaches dominate the trade-off, where ER and CPF maintain their superiority, while baseline tail exposure providing ample flexibility for re-ranking algorithms to redistribute exposure toward tail items without breaching the accuracy constrain.

When the recommendation list reaches $K = 50$ the baseline MMF jumps dramatically to 0.312, representing a four-fold increase from $K = 20$. This elevated baseline fundamentally alters the constraint landscape: the available wiggle room for additional fairness gains has shrunk greatly. Now several previously high-performing methods encounter the accuracy constraint as a hard ceiling. CPF achieves a high Relative MMF 1.178 but fails the budget with Relative NDCG = 0.987, below the 0.990 threshold, meaning its fairness gains are realized only at an unacceptable accuracy cost that exceeds the 1% loss allowance. ER similarly reaches a high Relative MMF of 1.173 but slightly violates the constraint Relative NDCG = 0.989, also exceeding the acceptable accuracy loss. Reg similarly breaches the budget with Relative NDCG = 0.987. The only post-processing method that survives the tightened constraint is P-MMF, which achieves a Relative MMF of 1.024 Relative NDCG = 1.000, and delivers the best secondary fairness metrics results with Gini = 0.181, Entropy = 0.904, EF = -0.344. Among the in-processing approaches, FOCF just barely satisfies the accuracy constraint with Relative NDCG at 0.990 with a Relative MMF of 1.025, while Reg again fails to meet the budget. At this cutoff, where baseline tail-exposure is already substantial, the admissible methods, P-MMF and FOCF, achieve nearly identical Rel-

ative MMF and secondary fairness scores. Critically, the fairness gains available to any method have collapsed to marginal improvements, with Relative MMF $\approx$ 1.02–1.03 versus 1.5–1.9 at smaller K, and the accuracy constraint now acts as a binding limitation rather than a loose guardrail.

The SASRec model on the Steam dataset reveals a different trade-off dynamic from the MovieLens findings, and the key differentiator is the relationship between baseline tail-exposure and available optimization flexibility. At smaller cutoff lengths ($K = 5, 10, 20$) where baseline MMF is extremely low (0.022–0.075), there exists substantial wiggle room for fairness interventions to redistribute exposure without triggering utility losses. In this regime, post-processing methods, particularly ER and CPF, consistently achieve the highest Relative MMF values (1.4–1.9 range) while maintaining comfortable compliance with the 1% NDCG budget. ER deserves particular emphasis as it achieves remarkable performance across all three smaller cutoffs. In-processing methods struggle to extract comparable fairness gains across these smaller cutoffs, suggesting that their direct regularization approach is less effective at exploiting the abundant optimization room afforded by the extreme baseline skewness.

At $K = 50$ where baseline MMF is already elevated at 0.312, the constraint landscape inverts: the available wiggle room for fairness improvements has compressed dramatically, and the accuracy ceiling becomes binding. Here, several post-processing methods (CPF, ER) that excelled at smaller K now violate the NDCG requirement because extracting further tail-exposure gains requires accuracy sacrifices that exceed the 1% threshold. Only the methods that achieve modest fairness improvements remain admissible. Thus, for SASRec on Steam, the effectiveness of fairness methodologies is fundamentally modulated by baseline tail-exposure levels. When baseline skewness is extreme, post-processing re-ranking methods dominate by leveraging abundant optimization flexibility; ER stands out as the most consistent top performer across the low-exposure regime, with CPF as a strong alternative. When baseline exposure disparity is already mitigated, both paradigms converge to marginal fairness gains constrained by the accuracy requirement. This finding stands in marked contrast to the MovieLens results, where in-processing methods dominated across all cutoffs, and suggests that the interplay between dataset skewness, model architecture, baseline fairness state, and optimization paradigm determines which methodology achieves the best accuracy–fairness trade-off.

**Table 4.7:** Performance Comparison and Trade-off Analysis for SASRec on the *Steam* Dataset. Results show performance for selected operating points across four cutoff lengths ($K = 5, 10, 20, 50$). Each configuration is optimized to maximize Relative MMF (Tail Exposure) subject to a utility budget constraint of **Relative NDCG** $\geq 0.990$ (max 1% loss relative to the baseline)

| Method | Paradigm | $K$ | Accuracy | | | Fairness | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | NDCG | RelNDCG | Recall | MMF | EF | Entropy | Gini | RelMMF |
| Baseline | — | 5 | 0.289 | – | 0.360 | 0.022 | -8.732 | 0.152 | 0.478 | – |
| CPF | Post | 5 | 0.286 | 0.993 | 0.361 | 0.042 | -4.635 | 0.250 | 0.458 | 1.900 |
| P-MMF | Post | 5 | 0.289 | 1.000 | 0.362 | 0.030 | -6.352 | 0.196 | 0.470 | 1.384 |
| ER | Post | 5 | 0.289 | 1.000 | 0.368 | 0.038 | -5.841 | 0.231 | 0.443 | 1.740 |
| FOCF | In | 5 | 0.289 | 1.005 | 0.367 | 0.022 | -8.685 | 0.153 | 0.478 | 1.005 |
| Reg | In | 5 | 0.284 | 0.986 | 0.361 | 0.030 | -6.481 | 0.193 | 0.470 | 1.356 |
| Baseline | — | 10 | 0.320 | – | 0.454 | 0.035 | -5.559 | 0.217 | 0.465 | – |
| CPF | Post | 10 | 0.317 | 0.991 | 0.452 | 0.054 | -3.550 | 0.304 | 0.446 | 1.562 |
| P-MMF | Post | 10 | 0.322 | 1.004 | 0.456 | 0.044 | -4.428 | 0.258 | 0.457 | 1.254 |
| ER | Post | 10 | 0.319 | 0.998 | 0.455 | 0.055 | -3.042 | 0.311 | 0.439 | 1.585 |
| FOCF | In | 10 | 0.319 | 0.998 | 0.463 | 0.041 | -4.754 | 0.245 | 0.460 | 1.167 |
| Reg | In | 10 | 0.317 | 0.991 | 0.458 | 0.043 | -4.465 | 0.257 | 0.457 | 1.245 |
| Baseline | — | 20 | 0.633 | – | 0.550 | 0.075 | -2.526 | 0.385 | 0.425 | – |
| CPF | Post | 20 | 0.344 | 0.991 | 0.547 | 0.109 | -1.694 | 0.496 | 0.391 | 1.445 |
| P-MMF | Post | 20 | 0.348 | 1.002 | 0.549 | 0.089 | -2.124 | 0.432 | 0.412 | 1.175 |
| ER | Post | 20 | 0.632 | 0.998 | 0.554 | 0.113 | -1.152 | 0.507 | 0.387 | 1.501 |
| FOCF | In | 20 | 0.345 | 0.994 | 0.555 | 0.084 | -2.257 | 0.415 | 0.416 | 1.112 |
| Reg | In | 20 | 0.343 | 0.990 | 0.553 | 0.091 | -2.067 | 0.439 | 0.409 | 1.206 |
| Baseline | — | 50 | 0.374 | – | 0.665 | 0.312 | -0.365 | 0.895 | 0.188 | – |
| CPF | Post | 50 | 0.367 | 0.987 | 0.662 | 0.368 | -0.227 | 0.949 | 0.132 | 1.178 |
| P-MMF | Post | 50 | 0.374 | 1.000 | 0.665 | 0.319 | -0.344 | 0.904 | 0.181 | 1.024 |
| ER | Post | 50 | 0.370 | 0.989 | 0.665 | 0.366 | -0.221 | 0.948 | 0.132 | 1.173 |
| FOCF | In | 50 | 0.368 | 0.990 | 0.662 | 0.320 | -0.344 | 0.904 | 0.180 | 1.025 |
| Reg | In | 50 | 0.367 | 0.987 | 0.662 | 0.325 | -0.330 | 0.910 | 0.175 | 1.042 |

### 4.3.2 Comparative Analysis of Fairness Trade-offs

#### Characterizing the Fairness-Utility Trade-off by Fairness Family and Cutoff

Table 4.8shows, for each recommendation cut-off $K$, the average relative utility (RelNDCG) and relative fairness (RelMMF) together with auxiliary metrics for the two fairness families (in-processing vs. post-processing) and the proportion of runs that satisfy the utility budget RelNDCG $\geq 0.990$.

It can be seen in Table 4.8 that In-processing and post-processing exhibit fundamentally different trade-off profiles across recommendation cutoffs. In-processing methods

consistently maintain the $\geq 0.990$ utility budget (87.5%–100% compliance), while post-processing methods struggle with this constraint, particularly at $K \in 5, 10, 20$ where compliance drops to 66.7%–75.0%. However, post-processing achieves substantially higher fairness gains (RelMMF of 1.401–1.632 at small $K$) compared to in-processing (RelMMF of 1.105–1.138 at small $K$).

At small cutoffs ($K = 5, 10$), post-processing methods exhibit a steep, costly trade-off: they gain 44%–47% relative fairness (RelMMF 1.632 vs. baseline) but with utility losses exceeding the 1% budget, with mean RelNDCG dropping to 0.965–0.971. This violates the optimization constraint and represents a costly outcome unsuitable for practitioners with strict accuracy requirements. In contrast, in-processing methods at the same cutoffs achieve a gentler trade-off: 13%–14% fairness gains (RelMMF 1.132–1.138) while maintaining $\approx 1.5\%$ utility surplus relative to the budget (mean RelNDCG 1.015–1.020).

As $K$ increases to 50, the fairness gains collapse for both families—post-processing achieves only 12.3% fairness gain (RelMMF 1.123) and in-processing only 4.2% gain (RelMMF 1.042). This saturation reflects the baseline saturation effect: as more items are ranked, the denominator in relative metrics grows, masking the absolute contribution of each method. At $K = 50$, post-processing improves its budget compliance to 75%, and mean RelNDCG rises to 0.992, suggesting that extreme fairness pushes are only feasible at large cutoffs where utility loss can be amortized across more items.

Secondary metrics (Entropy, Gini, EF) reinforce this narrative. At $K = 5$, post-processing achieves superior Entropy (0.689 vs. 0.599) and lower Gini (0.253 vs. 0.329), confirming higher diversity and fairness. In-processing lags on these tertiary fairness measures but maintains them at acceptable levels. By $K = 50$, the gap narrows substantially—both families converge toward Entropy $\approx 0.91$ and Gini $\approx 0.16$ reinforcing that large $K$ recommendations naturally ameliorate popularity bias regardless of intervention.

**Robustness to Recommender Architecture**

Table 4.9 reports for every method and cut-off $K$, the mean relative fairness (RelMMF) and relative utility (RelNDCG) obtained with the two base recommenders (SASRec and BPR) and the pooled standard deviation across the two models, which quantifies how sensitive each fairness intervention is to the choice of recommender architecture.

**Table 4.8:** Aggregated comparison grouped by $K$ (rows: metrics; columns: family). Budget: RelNDCG $\geq 0.990$.

| K | Metric | In-Processing | Post-Processing |
|---|---|---|---|
| 5 | Mean RelNDCG ↑ | 1.020 | 0.971 |
| | Mean RelMMF ↑ | 1.132 | 1.632 |
| | Mean EF ↑ | -2.497 | -1.751 |
| | Mean Entropy ↑ | 0.599 | 0.689 |
| | Mean Gini ↓ | 0.329 | 0.253 |
| | Budget Compliance ↑ | 87.5% | 75.0% |
| 10 | Mean RelNDCG ↑ | 1.015 | 0.965 |
| | Mean RelMMF ↑ | 1.138 | 1.522 |
| | Mean EF ↑ | -1.721 | -1.291 |
| | Mean Entropy ↑ | 0.627 | 0.698 |
| | Mean Gini ↓ | 0.321 | 0.252 |
| | Budget Compliance ↑ | 100.0% | 66.7% |
| 20 | Mean RelNDCG ↑ | 1.014 | 0.972 |
| | Mean RelMMF ↑ | 1.105 | 1.401 |
| | Mean EF ↑ | -1.049 | -0.761 |
| | Mean Entropy ↑ | 0.692 | 0.752 |
| | Mean Gini ↓ | 0.297 | 0.238 |
| | Budget Compliance ↑ | 100.0% | 66.7% |
| 50 | Mean RelNDCG ↑ | 1.019 | 0.992 |
| | Mean RelMMF ↑ | 1.042 | 1.123 |
| | Mean EF ↑ | -0.322 | -0.271 |
| | Mean Entropy ↑ | 0.913 | 0.925 |
| | Mean Gini ↓ | 0.171 | 0.158 |
| | Budget Compliance ↑ | 87.5% | 75.0% |

Table 4.9 shows that In-processing methods (FOCF and Reg) exhibit low variability across SASRec and BPR, with SD values typically $\leq 0.11$ for RelMMF and $\leq 0.03$ for RelNDCG across all cutoffs. FOCF is exceptionally stable: RelMMF ranges from 1.054–1.160 (SD $\leq 0.077$), and RelNDCG is consistently $\geq 1.000$ (SD $\leq 0.046$). This indicates that in-processing interventions impose fairness constraints at the model level in a way that is largely independent of whether the base architecture is a sequential model (SASRec) or matrix factorization (BPR). Practitioners can thus deploy in-processing methods with confidence that fairness–utility trade-offs will be predictable across model choices.

In stark contrast, post-processing methods demonstrate severe model sensitivity, especially P-MMF. At $K = 5$, P-MMF achieves RelMMF of 1.240 on SASRec but 2.289 on BPR (SD = 0.788), a 85% relative increase. More critically, this aggressive fairness push on BPR incurs a catastrophic utility loss: RelNDCG drops to 0.886 (exceeding

the 1% budget by a wide margin, resulting in an 11.4% utility loss). SASRec and BPR have fundamentally different item prediction distributions: sequential patterns in SASRec impose implicit ordering constraints that moderate aggressive re-ranking, while BPR's collaborative filtering approach allows re-ranking to more aggressively suppress popular items. P-MMF's unchecked optimization on BPR exploits this freedom, restulting extreme fairness gains but at high utility cost.

CPF and ER occupy a middle ground: SD values range from 0.214–0.340, indicating moderate model sensitivity. Both maintain better budget compliance than P-MMF (e.g., CPF at $K = 10$: RelNDCG 0.990 on SASRec, 0.939 on BPR) but still show base-model-dependent trade-offs. This suggests that CPF and ER apply gentler, more controlled re-ranking heuristics that partially adapt to the base model's ranking distribution, but lack the robustness of in-processing.

**Table 4.9:** Per-$K$ fair-model averages by family (Post vs. In) for SASRec vs. BPR, with Standard deviation across both basemodels.

| Method | Paradigm | $K$ | SASRec | | BPR | | SD | |
|---|---|---|---|---|---|---|---|---|
| | | | RelMMF | RelNDCG | RelMMF | RelNDCG | RelMMF | RelNDCG |
| CPF | Post | 5 | 1.502 | 0.994 | 1.656 | 0.980 | 0.340 | 0.052 |
| ER | Post | 5 | 1.426 | 0.998 | 1.680 | 0.975 | 0.296 | 0.045 |
| P-MMF | Post | 5 | 1.240 | 0.996 | 2.289 | 0.886 | 0.788 | 0.125 |
| FOCF | In | 5 | 1.070 | 1.000 | 1.080 | 1.059 | 0.077 | 0.046 |
| Reg | In | 5 | 1.282 | 1.004 | 1.098 | 1.019 | 0.139 | 0.019 |
| CPF | Post | 10 | 1.338 | 0.990 | 1.560 | 0.939 | 0.264 | 0.031 |
| ER | Post | 10 | 1.366 | 0.995 | 1.471 | 0.992 | 0.297 | 0.007 |
| P-MMF | Post | 10 | 1.144 | 1.002 | 2.250 | 0.870 | 0.763 | 0.125 |
| FOCF | In | 10 | 1.160 | 1.008 | 1.082 | 1.026 | 0.065 | 0.016 |
| Reg | In | 10 | 1.192 | 1.010 | 1.118 | 1.016 | 0.110 | 0.026 |
| CPF | Post | 20 | 1.251 | 0.990 | 1.428 | 0.962 | 0.282 | 0.025 |
| ER | Post | 20 | 1.294 | 0.996 | 1.400 | 0.986 | 0.296 | 0.009 |
| P-MMF | Post | 20 | 1.094 | 1.000 | 1.939 | 0.896 | 0.536 | 0.111 |
| FOCF | In | 20 | 1.108 | 1.011 | 1.068 | 1.010 | 0.025 | 0.016 |
| Reg | In | 20 | 1.176 | 1.012 | 1.069 | 1.020 | 0.068 | 0.021 |
| CPF | Post | 50 | 1.096 | 0.989 | 1.260 | 0.975 | 0.234 | 0.025 |
| ER | Post | 50 | 1.110 | 0.993 | 1.249 | 0.995 | 0.214 | 0.005 |
| P-MMF | Post | 50 | 1.012 | 1.001 | 1.012 | 0.997 | 0.010 | 0.004 |
| FOCF | In | 50 | 1.054 | 1.014 | 1.020 | 1.014 | 0.031 | 0.021 |
| Reg | In | 50 | 1.068 | 1.019 | 1.027 | 1.030 | 0.032 | 0.031 |

### Robustness to Dataset Characteristics

Table 4.10 presents the same set of metrics as Table 4.9 , but now the two columns correspond to the two benchmark datasets (ML-100k and Steam). The pooled standard deviation captures the variability of each method across domains that have markedly different sparsity and popularity-distribution characteristics.

Earlier in Table 4.1, it was shown that the two datasets present starkly different popularity distributions. ML-100k, with 38.92% of items classified as head (covering 80% of interactions) and sparsity 0.9220, exhibits a moderate, gradual long-tail distribution. Steam, conversely, with only 10.29% head items and sparsity 0.9985, exhibits an extreme, concentrated long-tail distribution. This difference manifests directly in the robustness profiles of fairness methods.

Table 4.10 shows that in-processing methods, FOCF and Reg maintain remarkably low SD across datasets. FOCF's RelMMF varies minimally: 1.073–1.090 on ML-100k vs. 1.076–1.153 on Steam (SD $\leq 0.045$), and RelNDCG stays tightly clustered around 1.01–1.045 (SD $\leq 0.022$). This robustness holds even though Steam's extreme sparsity provides far fewer training observations per user and item. In-processing methods decouple the fairness constraint from the base data distribution, allowing them to generalize across domains. The constraint is enforced during training via gradient signals or regularization, which are relatively insensitive to downstream sparsity variations.

Post-processing methods, by contrast, show high variability and systematic differences across domains. P-MMF again exemplifies this: RelMMF SD values range from 0.005 (K=50) to 0.526 (K=5), with the largest gaps at small $K$ where Steam requires much more aggressive re-ranking (RelMMF 2.136 on Steam vs. 1.392 on ML-100k at $K = 5$). Critically, this aggressiveness translates into severe utility penalties on Steam: RelNDCG drops to 0.877–0.896 at $K \in 5, 10, 20$, violating the budget by 12–13 percentage points.

Steam's extreme sparsity (0.9985 vs. 0.9220) and concentrated head (10.29% vs. 38.92% of items) create a more rigid popularity ranking. The head items are so dominant that the baseline recommendation list is highly skewed toward them. When post-processing methods attempt to promote tail items, they must perform severe rank inversions—moving tail items from, e.g., positions 100–500 into the top-20, which requires removing or demoting head items that users likely need. ML-100k's broader head allows more moderate re-ranking; tail items occupy better initial positions, re-

quiring gentler promotion. CPF and ER show moderate domain sensitivity (SD $\leq$ 0.247 for RelMMF), with systematic utility-fairness trade-off shifts. On Steam, both methods achieve lower RelNDCG than on ML-100k: e.g., CPF at $K = 5$ shows RelNDCG 1.019 on ML-100k but 0.955 on Steam. This 6.4 percentage point drop reflects the same mechanism: Steam's concentration forces a steeper re-ranking cost.

**Table 4.10:** Per-$K$ fair-model averages by family (Post vs. In) for MovieLens-100k vs. Steam, with Standard deviation across datasets.

| Method | Paradigm | $K$ | ML-100k | | Steam | | SD | |
|---|---|---|---|---|---|---|---|---|
| | | | RelMMF | RelNDCG | RelMMF | RelNDCG | RelMMF | RelNDCG |
| CPF | Post | 5 | 1.404 | 1.019 | 1.753 | 0.955 | 0.247 | 0.045 |
| ER | Post | 5 | 1.407 | 1.011 | 1.700 | 0.962 | 0.207 | 0.035 |
| P-MMF | Post | 5 | 1.392 | 1.005 | 2.136 | 0.877 | 0.526 | 0.091 |
| FOCF | In | 5 | 1.073 | 1.045 | 1.076 | 1.014 | 0.002 | 0.022 |
| Reg | In | 5 | 1.112 | 1.026 | 1.268 | 0.998 | 0.110 | 0.019 |
| CPF | Post | 10 | 1.424 | 0.970 | 1.474 | 0.959 | 0.035 | 0.008 |
| ER | Post | 10 | 1.450 | 0.996 | 1.388 | 0.992 | 0.045 | 0.003 |
| P-MMF | Post | 10 | 1.392 | 0.996 | 2.002 | 0.876 | 0.431 | 0.084 |
| FOCF | In | 10 | 1.090 | 1.027 | 1.153 | 1.007 | 0.045 | 0.014 |
| Reg | In | 10 | 1.072 | 1.035 | 1.238 | 0.991 | 0.117 | 0.031 |
| CPF | Post | 20 | 1.370 | 0.988 | 1.308 | 0.964 | 0.043 | 0.016 |
| ER | Post | 20 | 1.388 | 0.994 | 1.306 | 0.988 | 0.057 | 0.005 |
| P-MMF | Post | 20 | 1.346 | 1.004 | 1.688 | 0.892 | 0.242 | 0.079 |
| FOCF | In | 20 | 1.081 | 1.015 | 1.095 | 1.007 | 0.010 | 0.006 |
| Reg | In | 20 | 1.097 | 1.034 | 1.148 | 0.999 | 0.036 | 0.024 |
| CPF | Post | 50 | 1.261 | 0.998 | 1.095 | 0.966 | 0.118 | 0.022 |
| ER | Post | 50 | 1.264 | 0.998 | 1.094 | 0.990 | 0.121 | 0.006 |
| P-MMF | Post | 50 | 1.008 | 1.002 | 1.016 | 0.996 | 0.005 | 0.004 |
| FOCF | In | 50 | 1.051 | 1.030 | 1.023 | 0.999 | 0.020 | 0.022 |
| Reg | In | 50 | 1.062 | 1.050 | 1.032 | 0.999 | 0.022 | 0.036 |

RQ1 Answer (Trade-off Characterization):Post-processing methods achieve 3$\times$ higher fairness gains than in-processing at small $K$ (1.632 vs. 1.132 RelMMF at $K = 5$), but this comes at substantial and uncontrollable utility cost. In-processing methods follow a predictable, budget-compliant trade-off: 13–14% fairness gains for $\approx$ 1.5% utility loss at small $K$, with gains decaying gracefully to $\approx$ 4% at $K = 50$. For practitioners with strict accuracy constraints, in-processing is the only viable family. RQ2 Answer (Robustness):In-processing methods are robust across both architectures and domains: SD values remain $\leq$ 0.11 regardless of whether the base model is SASRec or BPR, or whether the data is moderately sparse (ML-100k) or extremely sparse (Steam).

Post-processing methods are architecture-sensitive and domain-sensitive, with P-MMF being particularly unstable when deployed on BPR or Steam, where it incurs utility penalties exceeding 10%. CPF and ER offer a middle ground but still exhibit model- and domain-dependent trade-offs. In-processing is the only method family suitable for practitioners who cannot afford to re-tune hyperparameters across new models or datasets.

# Chapter 5

# Discussion

Fairness aware models are essential for building responsible recommender systems, yet they inherently introduce a complex trade-off with recommendation accuracy. Existing research frequently evaluates these methods in isolation, hindering a quantitative understanding of how different method families—specifically in-processing versus post-processing, map the resulting fairness-utility frontier when held to a strict accuracy budget. Our research addresses this gap by establishing a unified, reproducible benchmark to investigate how these two method families affect item popularity-based fairness metrics (e.g., tail exposure) relative to user utility (NDCG). This chapter synthesizes the experimental results from this investigation, verifying the consistency of the observed trade-offs across different base recommenders (BPR and SASRec) and highly divergent data domains (MovieLens and Steam), thereby definitively answering the two primary research questions presented earlier.

## 5.1 RQ1: How do different fairness-aware methods (in-processing vs. postprocessing) shift the fairness–accuracy trade-off?

Post-processing methods consistently achieve larger gains on item-popularity-based fairness metrics (e.g., MMF, entropy) than in-processing methods. At small recommendation cut-offs ($K = 5, 10, 20$) the relative fairness improvement is markedly higher

for post-processing, but this comes at the cost of violating the strict utility budget (RelNDCG $\geq$ 0.990) for a substantial fraction of runs. In-processing methods, by contrast, preserve the utility constraint in the majority of configurations (=90% – 100% compliance) while delivering modest fairness improvements. As $K$ grows to 50, the advantage of post-processing diminishes: both families converge toward similar fairness levels and the utility budget is satisfied more frequently, reflecting the baseline-saturation effect in which larger recommendation lists naturally reduce popularity bias.

The different behaviour stems from where the fairness constraint is imposed. Post-processing operates on the final ranked list, allowing aggressive re-ranking that can dramatically reshuffle exposure toward tail items; however, such re-ordering inevitably displaces high-utility (often popular) items, producing the observed utility loss. In-processing embeds the fairness objective within the learning loss, so the model learns a more balanced representation but is limited by the capacity of the underlying recommender to trade accuracy for fairness. Consequently, in-processing results in a smoother trade-off that is more suitable when strict accuracy preservation is required. Post-processing offers a higher-risk, higher-reward option for practitioners who prioritize fairness and can tolerate modest utility loss.

These findings suggest a practical decision rule: if a deployment scenario enforces a tight NDCG budget (e.g., a production recommender with service-level accuracy guarantees), in-processing methods are the safer default. When the system can afford a modest utility loss, post-processing provides a more effective tool for boosting tail exposure and diversity.

## 5.2 RQ2: How consistent are these trade-offs across different base recommenders and across datasets from different domains?

Both fairness families behave consistently when the base model switches betweena sequential architecture (SASRec) and a classic matrix-factorisation model (BPR). In-processing methods exhibit very low variability in both relative fairness (RelMMF) and relative utility (RelNDCG) across the two models, indicating that the fairness regularisation learned during training is largely architecture-agnostic. Post-processing

methods, however, display markedly higher sensitivity:the same post-processing algorithm can result in very different fairness gains and utility losses depending on whether the base scores come from SASRec or BPR. This sensitivity arises because the two models produce distinct score distributions. BPR's collaborative-filtering scores are more responsive to aggressive rank reordering, which can amplify fairness gains but also exacerbate utility loss.

When moving from the moderate-sparsity MovieLens-100k dataset to the highly sparse Steam dataset, the overall fairness improvement changes dramatically. In-processing methods retain their modest but stable fairness gains and continue to respect the utility budget in both domains, demonstrating strong generalisation. Post-processing methods, in contrast, experience pronounced domain-dependent behaviour: on Steam they must perform far more aggressive rank inversions to promote tail items, leading to substantial drops in RelNDCG that breach the 1% utility budget. The variability is especially evident at small $K$, where the fairness boost on Steam is highest but comes with the steepest utility penalty.

When moving from the moderate-sparsity MovieLens-100k dataset to the highly sparse Steam dataset, the overallfairness improvement changes dramatically. The critical difference lies in popularity concentration. The Steam dataset, while defined by the standard 80/20 interaction distribution, achieves this definition with an extremely small percentage of the items classified as head items(only $\approx 10\%$ of items), whereas MovieLens distributes the same80% of interactions across a large head ($\approx 39\%$ of items). Post-processing methods, in particular, struggle with Steam's concentrated popularity, as promoting tail items requires highly disruptive rank inversions. This leads to them breaching the strict 1% utility budget more frequently and severely on Steam than on MovieLens. The observed drop in relative NDCG suggests that removing or demoting one of the few, hyper-popular head items in the Steam, results in a huge penalty. This creates a practical limit on how much we can promote tail items before the overall quality of the recommendations appears to suffer.

Overall, post-processing methods are less robust: their performance is highly contingent on both the underlying recommender's score distribution and the popularity structure of the dataset. In-processing methods generalise better, delivering predictable trade-offs across architectures and domains,albeit with lower absolute fairness gains. For practitioners who plan to deploy a fairness-aware system across heterogeneous platforms or datasets,in-processing offers a more reliable baseline; post-processing may still

be useful when a single, well-understood data domain permits aggressive re-ranking.

## 5.3 Limitations and Future Works

Our study is restricted to two public benchmarks, MovieLens-100k and Steam, because they deliberately exhibit opposite popularity distribution shapes (a highly concentrated head in Steam versus a more balanced tail in Movielens). This contrast was essential for answering our primary research question: does a fairness-aware recommender behave consistently across datasets with different head-size concentrations? Nevertheless, the conclusions may not generalize to domains with other characteristics such as strong temporal dynamics, multi-modal side information, or session-based interactions. Future work should extend the robustness analysis to additional corpora.

We selected two fundamentally different recommendation basemodels, BPR (a static matrix-factorisation model) and SASRec (a sequential transformer), to test whether fairness interventions are architecture-agnostic. While this choice directly supports our robustness inquiry, it leaves open the question of how the same fairness models interact with graph-neural, content-aware, or large-language-model recommenders. A natural extension is to replicate the experiments on graph-based models such as LightGCN or on LLM-driven recommenders, thereby mapping which fairness-aware techniques are best suited to each architectural family.

The experimental matrix (2 datasets$\times$ 2 base models $\times$ 5 fairness methods $\times$ 4 cut-off values) already demanded substantial GPU time; in-processing methods required full model retraining for each lambda, which limited our ability to add more base recommender model or datasets. Moreover, several recent fairness papers did not release source code and the authors did not respond to our requests, further curtailing the method set. Addressing these reproducibility and resource bottlenecks, will enable larger-scale robustness studies in the future.

# Chapter 6

# Conclusion

In this research project, we established a unified and reproducible benchmark to investigate the core trade-off between fairness and accuracy in recommender systems. Our study quantitatively compared two major families of fairness interventions—in-processing regularization and post-processing re-ranking—and systematically evaluated the robustness of the resulting fairness–utility frontiers across diverse base models and datasets. Our investigation focused on addressing two primary research questions: (RQ1) how do different fairness-aware methods shift the fairness–accuracy trade-off? and (RQ2) how consistent are these trade-offs across different base recommenders and datasets? To answer RQ1, we found a fundamental divergence in the behavior of the two method families. Post-processing consistently yielded higher absolute gains in item-popularity-based fairness metrics such as MMF (tail exposure) but frequently breached the strict utility budget (RelNDCG $\geq 0.990$) at small recommendation cut-offs. Conversely, in-processing reliably preserved the utility constraint, providing a modest, fairness gain.

The robustness analysis addressed in RQ2 demonstrated that in-processing methods are much more stable. They produce similar fairness gains and utility losses whether we use a sequential model (SASRec) or a matrix-factorisation model (BPR), and they behave consistently on both the MovieLens and Steam datasets. In contrast, post-processing methods are far more sensitive to the underlying recommender's score distribution. On the highly concentrated Steam data, they have to reorder the ranking

list aggressively to promote tail items, which often leads to a noticeable drop in NDCG and exceeds the 1% utility-budget limit.

Due to resource limitations and issues with external code availability, our experimental matrix was constrained to two datasets and two base models. Future studies should replicate this methodology on emerging architectures such as Graph Neural Networks and Large Language Model-driven recommenders. Furthermore, extending the robustness analysis to a wider array of corpora would facilitate the development of an automated framework capable of selecting the optimal fairness intervention based on a dataset's innate characteristics.

# Chapter 7

# Bibliography


[1] Himan Abdollahpouri and Robin Burke. Multi-stakeholder recommendation and its connection to multi-sided fairness. *arXiv preprint*, 2019.

[2] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H. Chi, and et al. Fairness in recommendation ranking through pairwise comparisons. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD '19)*, pages 2212–2220, 2019.

[3] Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. Equity of attention: Amortizing individual fairness in rankings. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 405–414, 2018.

[4] Robin Burke. Multisided fairness for recommendation. *arXiv preprint*, 2017.

[5] Carlos Castillo. Fairness and transparency in ranking. *ACM SIGIR Forum*, 52(2):64–71, 2019.

[6] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. Bias and debias in recommender system: A survey and future directions. *arXiv preprint arXiv:2010.03240*, 2020.

[7] Xiao Chen, Wenqi Fan, Jingfan Chen, Haochen Liu, Zitao Liu, Zhaoxiang Zhang, and Qing Li. Fairly adaptive negative sampling for recommendations. In *Proceedings of the ACM Web Conference 2023 (WWW)*, pages 3723–3733, 2023.

[8] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogin, Alessandro Difonzo, and Dario Zanzonelli. Fairness in recommender systems: research landscape and fu-

ture directions. *User Modeling and User-Adapted Interaction*, 34(1):59–108, April 2023.

[9] Michael D. Ekstrand, Mucun Tian, Ion Madrazo Azpiazu, Jennifer D. Ekstrand, Oghenemaro Anuyah, David McNeill, and Maria Soledad Pera. All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness. In Sorelle A. Friedler and Christo Wilson, editors, *Proceedings of the 1st Conference on Fairness, Accountability and Transparency (FAT*)*, volume 81 of *Proceedings of Machine Learning Research*, pages 172–186, New York, NY, 2018. PMLR.

[10] Boli Fang, Miao Jiang, Pei-Yi Cheng, Jerry Shen, and Yi Fang. Achieving outcome fairness in machine learning models for social decision problems. In *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI)*, pages 444–450, Virtual Conference, 2020. IJCAI.

[11] Meng Fang, Jin Liu, Masashi Momma, and Yizhou Sun. Fairroad: Achieving fairness for recommender systems with optimized antidote data. In *Proceedings of the 27th ACM Symposium on Access Control Models and Technologies (SACMAT)*, 2022.

[12] Michael Färber, Melissa Coutinho, and Shuzhou Yuan. Biases in scholarly recommender systems: Impact, prevalence, and mitigation. *Scientometrics*, 128(5):2703–2736, 2023.

[13] Carlos A. Gómez-Uribe and Neil Hunt. The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems*, 6(4):13:1–13:19, 2015.

[14] Dietmar Jannach, Pearl Pu, Francesco Ricci, and Markus Zanker. Recommender systems: Trends and frontiers. *AI Magazine*, 43(2):145–150, 2022.

[15] Di Jin, Luzhi Wang, He Zhang, Yizhen Zheng, Weiping Ding, Feng Xia, and Shirui Pan. A survey on fairness-aware recommender systems. *Inf. Fusion*, 100(C), December 2023.

[16] Joseph M. Juran. *Quality-Control Handbook*. McGraw-Hill, New York, NY, USA, 1951.

[17] Toshihiro Kamishima and Shotaro Akaho. Considerations on recommendation independence for a find-good-items task. In *Workshop on Responsible Recommendation (FATREC) in conjunction with RecSys 2017*, pages 1–6, Como, Italy, 2017.

[18] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *Proceedings of the 2018 IEEE International Conference on Data Mining*

*(ICDM)*, pages 197–206, 2018.

[19] Min Kyung Lee, Anuraag Jain, Hea Jin Cha, Shashank Ojha, and Daniel Kusbit. Procedural justice in algorithmic fairness: Leveraging transparency and outcome control for fair algorithmic mediation. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW), November 2019.

[20] Roger Zhe Li, Julián Urbano, and Alan Hanjalic. Leave no user behind: Towards improving the utility of recommender systems for non-mainstream users. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining (WSDM '21)*, pages 103–111, New York, NY, 2021. ACM.

[21] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, Juntao Tan, Shuchang Liu, and Yongfeng Zhang. Fairness in recommendation: Foundations, methods, and applications. *ACM Transactions on Intelligent Systems and Technology*, 14(5):95:1–95:48, 2023.

[22] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, Juntao Tan, Shuchang Liu, and Yongfeng Zhang. Fairness in recommendation: Foundations, methods, and applications. *ACM Trans. Intell. Syst. Technol.*, 14(5), October 2023.

[23] Martin Mladenov, Elliot Creager, Omer Ben-Porat, Kevin Swersky, Richard Zemel, and Craig Boutilier. Optimizing long-term social welfare in recommender systems: A constrained matching approach. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pages 6987–6998. PMLR, 2020.

[24] Mohammadmehdi Naghiaei, Hossein A. Rahmani, and Yashar Deldjoo. CPFair: Personalized consumer and producer fairness re-ranking for recommender systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 770–779, 2022.

[25] Mohammadmehdi Naghiaei, Hossein A. Rahmani, and Yashar Deldjoo. Personalized consumer and producer fairness re-ranking for recommender systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '22)*, Madrid, Spain, 2022. ACM.

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

[27] Gourab K. Patro, Arpita Biswas, Niloy Ganguly, Krishna P. Gummadi, and Abhijnan Chakraborty. Two-sided fairness for personalized recommendations in two-sided platforms. In *Proceedings of the 2020 WWW Conference (The Web Conference)*, page 11 pages, Taipei, Taiwan, 2020. ACM.

[28] Bashir Rastegarpanah, Krishna P. Gummadi, and Mark Crovella. Fighting fire with fire: Using antidote data to improve polarization and fairness of recommender systems. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 231–239, 2019.

[29] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 452–461, 2009.

[30] Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors. *Recommender Systems Handbook*. Springer, Boston, MA, 2011.

[31] Tianhao Shi, Yang Zhang, Jizhi Zhang, Fuli Feng, and Xiangnan He. Fair recommendations with limited sensitive attributes: A distributionally robust optimization approach. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '24)*, page 10 pages, Washington, DC, USA, 2024. ACM.

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

[33] Yifan Wang, Weizhi Ma, Min Zhang, Yiqun Liu, and Shaoping Ma. A survey on the fairness of recommender systems. *ACM Transactions on Knowledge Discovery from Data*, 16(6):92:1–92:41, 2022.

[34] Chuhan Wu, Fangzhao Wu, Xiting Wang, Yongfeng Huang, and Xing Xie. Fairness-aware news recommendation with decomposed adversarial learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4462–4469. AAAI Press, 2021.

[35] Chen Xu, Sirui Chen, Jun Xu, Weiran Shen, Xiao Zhang, and Gang Wang. Provider max-min fairness re-ranking in recommender system. In *Proceedings of the ACM Web Conference (WWW '23)*, Austin, TX, USA, 2023. ACM.

[36] Chen Xu, Sirui Chen, Jun Xu, Weiran Shen, Xiao Zhang, Gang Wang, and Zhenghua Dong. P-MMF: Provider max-min fairness re-ranking in recommender system. In *Proceedings of the ACM Web Conference 2023 (WWW)*, pages 3701–3711, 2023.

[37] Chen Xu, Jun Xu, Yiming Ding, Xiao Zhang, and Qi Qi. Fairsync: Ensuring amortized group exposure in distributed recommendation retrieval. In *Proceedings of the ACM Web Conference 2024 (WWW '24)*, pages 13–24, Singapore, 2024. ACM.

[38] Chen Xu, Xiaopeng Ye, Wenjie Wang, Liang Pang, Jun Xu, and Tat-Seng Chua. A taxation perspective for fair re-ranking. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1494–1503, 2024.

[39] Chen Xu, Jujia Zhao, Wenjie Wang, Liang Pang, Jun Xu, Tat-Seng Chua, and Maarten de Rijke. Understanding accuracy-fairness trade-offs in re-ranking through elasticity in economics. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2025.

[40] Chen Xu, Jujia Zhao, Wenjie Wang, Liang Pang, Jun Xu, Tat-Seng Chua, and Maarten de Rijke. Understanding accuracy-fairness trade-offs in re-ranking through elasticity in economics. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25)*, Padua, Italy, 2025. ACM.

[41] Sirui Yao and Bert Huang. Beyond parity: Fairness objectives for collaborative filtering. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, pages 2921–2930, 2017.

[42] Xiaopeng Ye, Chen Xu, Jun Xu, Xuyang Xie, Gang Wang, and Zhenhua Dong. Guaranteing accuracy and fairness under fluctuating user traffic: A bankruptcy-inspired re-ranking approach. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management (CIKM '24)*, page 11 pages, Boise, ID, USA, 2024. ACM.

[43] Ziwei Zhu, Jianling Wang, and James Caverlee. Measuring and mitigating item under-recommendation bias in personalized ranking systems. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 449–458, 2020.