Fairness Implications of Data Minimization in Deep Collaborative Filtering

Sipei Li
Sipei.Li@tufts.edu
Tufts University
Medford, MA, USA

Nasim Sonboli
nasim_sonboli@brown.edu
Brown University
Providence, RI, USA

Mehdi Elahi
Mehdi.Elahi@uib.no
MediaFutures, University of Bergen
Bergen, Norway

Alva Couch
Alva.Couch@tufts.edu
Tufts University
Medford, MA, USA

Abstract

Data Minimization, a core principle of the General Data Protection Regulation (GDPR), requires limiting personal data "[...] to the purpose for which they are processed." However, there is still not a clear definition of data minimization, and indeed, its algorithmic implications for machine learning remain insufficiently understood. This gap is particularly notable in the research area of Recommender Systems (RS). RS rely on large-scale data collection and processing. It remains unclear how data minimization should be implemented in such models. This is particularly important since any limitation on data may affect accuracy and system fairness, due to disproportionate data processing across different user groups. In this paper we study the practical implications of data minimization in RS. We analyze the performance of RS when operationalizing data minimization via Active Learning (AL). A set of commonly-used AL strategies are implemented and then thorough empirical evaluations are conducted on them with respect to accuracy and fairness. To generate recommendations, we use a popular type of RS, namely deep Collaborative Filtering, which utilizes state-of-the-art deep learning methods to learn from user data. Our results demonstrate that depending on the type of RS, certain AL strategies are able to improve the model performance to a greater extent. Nonetheless, all the AL strategies negatively affect fairness, leading to trade-offs in implementing data minimization for RS.

Introduction

Regulatory frameworks such as the European Union's General Data Protection Regulation and the California Privacy Rights Act are increasingly shaping how data-driven systems are built and operated. These regulations establish a framework of principles, such as data minimization, accuracy, transparency, purpose limitation, and storage limitation, that govern the collection and processing of personal data. We focus on Data Minimization (DM), which GDPR Article 5(1)(c) defines as collecting and processing only data that is "adequate, relevant, and limited" to what is necessary for the stated purpose. Although different research communities have proposed techniques to implement DM, how to interpret the principle and implement it reliably in practice requires further investigation. For example, the impact of DM on system accuracy and its trade-off with other GDPR principles — especially fairness — is not yet sufficiently explored. This is critical since limitations on data collection can disproportionately impact different user groups, therefore potentially increasing unfairness.

Recommender systems (RS) learn from large-scale user interaction data to deliver to users contents tailored to their preferences. Modern RS often use Active Learning (AL) to selectively obtain high-quality data while reducing overall collection. This mechanism is particularly beneficial during the early stages of RS operation, enabling efficient user profiling and enhanced personalization (Beheshti et al. 2022).

In this work, we operationalize DM in movie RS using active learning (AL). AL identifies only the most useful data (e.g., movie items) and query users for their feedback (e.g., ratings or clicks). The collected feedback data is then utilized by the system to learn user preferences and generate recommendations. In this sense, AL strategies can serve the purpose of DM by identifying potentially the most useful interaction data points and letting the users to voluntarily share them with the system. AL thus limits the collection of user interaction data, which has been shown to enable the identification of user profiles (Narayanan and Shmatikov 2008). However, the usage of AL strategies — or any DM technique — may have impacts on fairness, since uneven data distributions can intensify disparities in the quality of recommendations across user populations. For example, strategies that prioritize items more familiar to specific user groups may improve model performance for those users, while making the system less useful to others.

Here, we analyze RS trained on data obtained via AL strategies, where model performance is highly sensitive to the quantity and quality of user interaction data. We consider not only the recommendation accuracy, but also consumer-side and provider-side fairness. Accordingly, we aim to address the following research questions:

- **RQ1:** How do AL strategies, when used as DM techniques, affect the accuracy of recommendations?
- **RQ2:** Do different AL strategies introduce or amplify

consumer-side or provider-side unfairness in recommendations?

In addressing these questions, we design and conduct experiments in which simulated users register with a RS and are presented with a set of items to provide feedback for. The items are selected using various AL strategies, and the collected feedback data are subsequently used to train the RS. We consider a range of AL strategies, including *Highest Predicted*, *Highest-Lowest Predicted*, and *Popularity*. We also include *Random* as a baseline strategy. For recommendation generation, we focus on state-of-the-art deep collaborative filtering (CF) approaches and evaluate them, i.e., NeuMF, LightGCN, and MultiVAE.

Our results demonstrate that the AL strategies can effectively improve model performance under limitations on data collection, while also emphasizing that the choice of the AL strategy should depend on the type of deep CF model being used. Results on consumer-side unfairness indicate the presence of performance disparities between user groups during the AL process, revealing a clear trade-off between model performance and consumer-side fairness. In addition, certain strategy is better at achieving a balance between model performance and provider-side fairness, while all the AL strategies negatively impact provider-side fairness in the early stage. Overall, these findings illustrate the practical challenges of operationalizing DM via AL, with a highlight on the importance of considering fairness when designing AL strategies for recommendation models.

Related Work

Data Minimization

Designing AI algorithms that are compliant with the DM principle is challenging because the legal requirements are not always straightforward to interpret, the principle lacks a standard mathematical formulation, and practical guidance is limited (Finck and Biega 2021). DM is defined by three principles: adequacy, relevance, and purpose limitation, which can be mapped into practical considerations in AI. *Adequacy* limits how much data a model collects or uses, *relevance* requires that the retained data contribute to the stated purpose, commonly assessed through model performance (e.g., accuracy), and *purpose limitation* restricts collection and processing to what is "necessary" for that goal. This translation provides the basis for prior work that proposes methods to operationalize DM, examines how it can be integrated into AI, and evaluates its impact on model performance. In this work, we analyze how DM constraints affect fairness in RS.

Several studies examine how this tension appears in RS. Biega et al. (2020) propose two performance-based formulations of DM (global and per-user) and show that recommendation quality can be maintained with less data, but that the impact of DM varies across users and may disproportionately reduce performance for minority groups. Shanmugam et al. (2022) introduce FIDO, which uses performance curves to set stopping criteria for data collection during training and demonstrates that additional data does not always improve per-user accuracy, especially when collected data is not representative. Galdon Clavell et al. (2020) audit a health RS (REM!X) under DM by not collecting sensitive attributes. They maintain accuracy but cannot assess or audit group fairness without those attributes. Concluding that some personal data may be necessary, especially if the lack of it causes inaccuracy or unfairness. Therefore, DM should be applied alongside GDPR principles such as fairness. Finck and Biega (2021) analyze how GDPR's data minimization and purpose limitation apply to personalization and profiling, highlighting practical challenges and trade-offs and motivating further research on minimization–fairness tensions.

More recently, Sonboli et al. (2024) implement DM in CF via AL and show that DM interventions can worsen group fairness. However, their empirical analysis, the model choice, and evaluation are limited in scope. Bufi et al. (2025) present an empirical auditing study of DM in RS, showing that DM can harm provider fairness while sometimes improving group-based consumer fairness, often at the cost of lower accuracy. They reinforce the need to evaluate minimization through a multi-objective fairness

Active Learning

Research on AL in RS is already an established direction, and many AL strategies have been proposed and evaluated in prior work. These strategies can be either personalized or non-personalized, each offering various advantages (Elahi, Ricci, and Rubens 2014). The personalized category of strategies requires analyzing users' prior ratings and building personalized models that ask different users to rate different sets of items. The advantage of this group of strategies is that accounting for individual user preferences during the AL process may provide a more engaging user experience while remaining informative for the RS. However, these models are typically complex and therefore computationally expensive. Non-personalized strategies, on the other hand, usually ignore users' prior ratings and query all users to rate the same items. Hence, they do not require complex models and are more efficient in the runtime. However, they may select items that do not necessarily match user preferences and, may appear obscure or unfamiliar, potentially impacting user engagement in a negative manner.

Examples of personalized strategies are prediction-based strategies, which first build a predictive model from elicited user preferences to predict future preferences and identify appropriate items for rating. For instance, the Highest Predicted strategy (Elahi, Repsys, and Ricci 2011) ranks items by predicted ratings and selects the top items. These strategies present users with items predicted to be relevant or familiar, increasing the chance of informative feedback. Examples of a non-personalized strategies is Popularity (He, Liu, and Yang 2011; Rashid et al. 2002), presenting same popular items to all users for rating, providing a straightforward yet limited form of exploration. It is often used as a baseline strategy (Golbandi, Koren, and Lempel 2010).

Methodology

In this section, we describe the methodology used to conduct the experiments in this paper.

Recommendation Models

We first provide a brief description for each of the deep CF models that are considered.

Neural Matrix Factorization (NeuMF) combines the linearity of Matrix Factorization and the non-linearity of a Multi-Layer Perceptron (MLP) to model the interactions between user and item embeddings. (He et al. 2017)

LightGCN is a simplification of Neural Graph Collaborative Filtering (Wang et al. 2019) that incorporates the interaction history information of a user or an item into a latent representation via multi-layer neighborhood aggregations using the interaction graph. (He et al. 2020)

MultiVAE uses an encoder-decoder structure to learn latent user representations from their interaction histories and makes predictions by using the learned representations to define a probability distribution over the items. (Liang et al. 2018)

Active Learning Strategies

In the AL process, the united system of an AL strategy and a RS (called *the system* thereafter) keeps track of a dataset of explicit or implicit feedback, recording every user's interaction data that they consent to share with the system. This dataset, referred to as the *known set*, is updated when the system queries each user about a list of items that they have not interacted with yet, and the users provide feedback by selectively interacting with some of the query items. The known set is used to train the RS to make recommendations. It is also used by the AL strategy to generate lists of query items in the next step. We only consider implicit feedback data in this work since most of the deep CF models considered here are designed for such data.

Let M and N denote the number of users and items. The known set can be represented as a binary matrix $Y \in \{0,1\}^{M\times N}$, which is often very sparse. The goal of the RS is to predict the missing values in Y and for each user, recommend the top items that are most probable to be of that user's interest.

The system starts the AL process with an initial known set $Y^{(0)}$. At each step t in the process, the AL strategy first outputs, for each user u, a set of items, i.e., the *query list* $Q_u^{(t)}$. The query list is a function:

$$Q_u^{(t)} = S(u, q, Y^{(t)}, P_u^{(t)}), \tag{1}$$

where q is a pre-defined maximum size of query lists, $Y^{(t)} \in \mathbb{R}^{M\times N}$ is the known set at step t, and $P_u^{(t)}$ is a pool of items that has not been queried for user u until step t. The query list $Q_u^{(t)}$ is a subset of $P_u^{(t)}$. Once an item is queried for user u, it is removed from $P_u^{(t)}$:

$$P_u^{(t+1)} = P_u^{(t)} \setminus Q_u^{(t)}, \tag{2}$$

where $|Q_u^{(t)}| = \min\{q, |P_u^{(t)}|\}$, with $|\cdot|$ denoting the size of a set.

After the query lists $Q_u^{(t)}$ have been generated, each user u chooses to interact with (e.g., click) the items in $Q_u^{(t)}$ that are not only of their interests but also the ones whose interactions they are willing to share with the system. The set of interacted items, i.e., the *solicited list*, $E_u^{(t)} \subseteq Q_u^{(t)}$, are then used to update the known matrix $Y^{(t)}$:

$$y_{ui}^{(t+1)} = 1, \quad \forall i \in E_u^{(t)}. \tag{3}$$

Note that a user u may not interact with any of the item in $Q_u^{(t)}$. In step $t+1$, the system repeats everything with the updated $Y^{(t+1)}$ and $P_u^{(t+1)}$.

The definition of the function S varies between AL strategies, as different heuristics are leveraged to estimate the contributions of the data to be collected. As mentioned in the Related Work section, these strategies can be categorized into two classes: personalized and non-personalized strategies. Personalized strategies utilize individual users' previously solicited interactions to generate user-specific query lists. We focus on the *prediction-based* personalized strategies that employ the RS trained on $Y^{(t)}$ to rank the items.

Highest Predicted scores each item i in $P_u^{(t)}$ using the RS output $\hat{y}_{ui}$. The q items with the highest scores are included in $Q_u^{(t)}$. Highest Predicted assumes that $\hat{y}_{ui}$ correctly reflects the possibility that user u will interact with item i. This strategy is the most straightforward and widely-used in real RS. (Elahi, Ricci, and Rubens 2016)

Highest-lowest Predicted is a *combined-heuristic* personalized strategy that selects a mixture of $q/2$ items with highest and $q/2$ with lowest model predictions $\hat{y}_{ui}$. It is argued by Karimi et al. (2011) that, especially for cold users, items with low prediction scores could be a result of suboptimal model parameters, and collecting interactions for such items could be beneficial for the model.

Non-personalized strategies use item selection rules based on item heuristics summarized from the overall known matrix $Y^{(t)}$ and query all the users for the same list of q items.

Popularity is an instance of *attention-based* non-personalized strategies. The q most popular items, i.e., those with the most number of interactions in $Y^{(t)}$, are selected, since the users are most likely to interact with them. However, Popularity suffers from the problem of prefix bias (Rashid et al. 2002) and sometimes may not provide useful information to the system.

Random is a simple non-personalized strategy where the q items are randomly selected from all the items that have not been queried for yet. To increase the probability of collecting at least one interaction in Random, we replace one item with an item randomly selected from a set of popular item in $Y^{(t)}$. Due to its randomized nature, Random is used as the baseline strategy in our experiments.

Experimental Setup

Dataset

We conduct experiments on the MovieLens-1M dataset (Harper and Konstan 2015), a public movie recommendation dataset comprising integer ratings in the range of 1 to 5, from 6,040 users on approximately 3,900 movies. The dataset is cleaned by iteratively filtering out users and items with less than 25 interactions, resulting in an interaction matrix consisting of 5,536 users and 2,910 items with a density

of 6.1%. Mean of number of ratings from each user is 177, and median is 106.

As mentioned in the Methodology section, the deep CF models are trained using implicit feedback data. Therefore, all the ratings are converted to 1's to encode implicit feedback. The dataset is split into training and test sets in a *user-fixed* manner, where 80% of each user's feedback data is included in the training set and 20% in the test set.

We use the Gender and Age attributes provided by the MovieLens-1M dataset to define the user groups in the consumer-side unfairness metrics mentioned in the next section. When the users are grouped by the Gender attribute, the number of users in the female group and male group are 1,538 and 3,998, and the respective average number of interactions are 156 and 185. When the users are grouped by the Age attribute, the number of users in the over-45 group and the under-45 group are 1,277 and 4,259, and the respective number of interactions are 148 and 186.

Evaluation Metrics

Fairness evaluation in RS is treated as a multi-stakeholder problem (Burke et al. 2024; Sonboli et al. 2022; Smith et al. 2025; Sonboli et al. 2020), where outcomes should be assessed for one or more of users/consumers, item/providers, and the platform. Surveys by Wang et al. (2023) and Deldjoo et al. (2024) synthesize the space of fairness definitions and argue for reporting multiple notions of fairness rather than a single score.

To measure the consumer-side unfairness in the models, we adopt the user group-level disparity, also known as the Mean Absolute Difference, in the performance of the RS: (Yao and Huang 2017):

$$\Delta\mathcal{M}(Z_1, Z_2) = \left| \frac{1}{|Z_1|} \sum_{u \in Z_1} \mathcal{M}(R_u) - \frac{1}{|Z_2|} \sum_{u \in Z_2} \mathcal{M}(R_u) \right|, \tag{4}$$

where Z_1 and Z_2 are two user groups defined by some user attribute (i.e., Gender and Age groups in our experiments), R_u is the top-k recommendation list for user u, and $\mathcal{M}$ is a metric of the quality of R_u. We use two ranking metrics for $\mathcal{M}$: NDCG@k and Precision@k, where k is set to 10. A higher value in $\Delta\mathcal{M}$ indicates a larger disparity in the recommendation quality across user groups.

For provider-side unfairness, we use Ranking-based Statistical Parity (RSP@k) and Ranking-based Equal Opportunity (REO@k) (Zhu, Wang, and Caverlee 2020), using the item categories provided by the dataset (i.e., movie genres). RSP@k is defined as the relative standard deviation of $P(\text{Rec@}k|g = g_a), a = 1, \dots A$, the probabilities of items in each category to be included in the top-k recommendation lists. And REP@k is defined as the relative standard deviation of $P(\text{Rec@}k|g = g_a, y = 1), a = 1, \dots, A$, the probabilities of items in each category to be included in the top-k recommendation lists given that the items are actually liked by the users (i.e., present the test set). For both RSP@k and REO@k, a higher value means there is a higher imbalance between item categories in the probability of being recommended. These metrics measure a crucial aspect of recommendation algorithms, i.e., the exposure-based bias, where certain item categories receive higher exposure rates. Such bias can in turn create feedback loops that further suppress the exposure of other item categories, exacerbating the bias in data and fostering filter bubbles. (Chen et al. 2023)

Experiment Procedure

We monitor the performance and unfairness metrics of the deep CF models trained using data collected by each of the AL strategies.

To simulate user behaviors in an offline experiment, we use the training set, denoted as $\tilde{Y}$, as an *oracle*. When user u is queried for item i, we simulate whether the u provides feedback for i according to whether the value of $\tilde{Y}_{ui}$ is 1. Note that we are using the training set to simulate not only the users' preferences on the items, but also the users' willingness to share the interactions with the system. It is also worth pointing out that due to the finite size of $\tilde{Y}$, it will eventually be fully recovered in the known set $Y^{(t)}$, when t is large enough.

For an AL strategy and a RS, we start by setting every $P_u^{(0)}$ to be the set of all items. Then we randomly sample 5% of the $\tilde{Y}$ as the initial known set $Y^{(0)}$ and remove the items from the corresponding $P_u^{(0)}$. Each step t of the AL process consists of:

1. Train the RS using $Y^{(t)}$;
2. Evaluate the performance of the trained RS on the test set, as well as the unfairness metrics, following Eq. 4 and definitions of RSP@k and REO@k;
3. Generate the query list $Q_u^{(t)}$ for each user u according to Eq. 1, and update $P_u^{(t)}$ according to Eq. 2;
4. Collect implicit feedback from the candidate set using the query lists, and then update $Y^{(t)}$, following Eq. 3.

Repeat until we reach a pre-defined number of steps T or the candidate set is exhausted. We use $q = 20$ and $T = 30$.

Reproducibility Details on Model Structures

For the NeuMF model, we use embeddings of size 32, 3 hidden layers in MLP, 4 negative samples per positive sample, a learning rate of 0.001, and 10 training epochs.

For LightGCN, we use embeddings of size 64, 2 layers of propagation, 1 negative sample per positive sample, a learning rate of 0.005, a weight decay of 0.001, and 20 training epochs.

For MultiVAE, we use latent vectors of size 70, one hidden layer of size 200, a dropout rate of 0.5, a learning rate of 0.001, and 200 training epochs.

Results and Discussion

Model Performance Under AL Strategies (RQ1)

In this section, we discuss the impact of AL strategies, as DM techniques, on the performance of deep CF models. We first note that since AL collects data from users in an *interactive* manner, there are two ways to interpret the minimization of data collection:

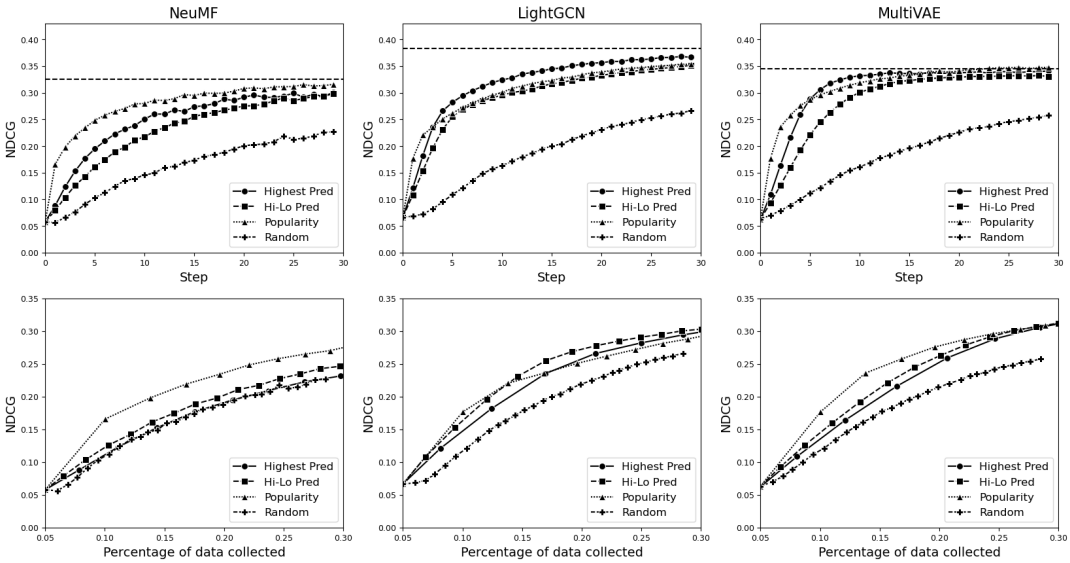

Figure 1: (Top) Average test set NDCG@10 of the CF models vs. number of AL steps. The horizontal dashed lines show the performance of the models trained on the whole training set. Highest Predicted, Highest-Lowest Predicted, and Popular are able to increase the model performance faster than Random. (Bottom) Average test set NDCG@10 of the CF models vs. the percentage of data collected from the overall training set. Highest Predicted, Highest-Lowest Predicted, and Popular are generally able to gain more model performance compared to Random-Popular, when collecting the same amount of data. Only exception is with NeuMF, where Highest Predicted gets similar model performance as Random.

1. The number of query items is limited, and we allow the users to decide how many of the query items they want to share with the system.
2. The number of collected feedback data is limited, as in other non-interactive DM techniques.

We start by placing the AL strategies in the context of the first interpretation and investigate their impact on the model performance. The top row of Figure 1 illustrates the average test set performance of the deep CF models when different AL strategies have been employed for a certain number of steps. We only include the results for NDCG@10 due to space limitation. However, we have observed similar trends for Precision@10.

As shown in the top row of Figure 1, for all the deep CF models, the strategies Highest Predicted, Highest-Lowest Predicted, and Popularity achieve better model performance than the Random baseline at any step of the AL process. This means that given a limitation on the number of query items (i.e., number of AL steps), these strategies are able to obtain better recommendation accuracy than the baseline, demonstrating their effectiveness as DM techniques under the first interpretation of data collection minimization.

The relative results of these strategies, however, vary for different CF models. For NeuMF, Popularity achieves the best model performance at all the AL steps, followed by Highest Predicted. But with LightGCN and NeuMF, Popularity stays the best for the first few steps and is then replaced by Highest Predicted, while Highest-Lowest gradually gets closer to Popularity. We will briefly discuss the possible cause of this difference later.

The bottom row of Figure 1 places the AL strategies in the context of the second interpretation of data collection minimization. Here, we plot the average test set performance of the deep CF models when they are trained on a certain amount of data, calculated as the percentage $Y^{(t)}/\tilde{Y}$. It can be seen that for LightGCN and MultiVAE, Highest Predicted, Highest-Lowest Predicted, and Popularity all have a higher performance-to-data ratio than the baseline. This means that under a limitation on the overall amount of collected data, these AL strategies are able to collect feedback data with higher quality, attaining better model performance. For NeuMF, Popularity and Highest-Lowest Predicted still exhibit better model performance than the baseline, but the difference between Highest Predicted and the baseline is not significant. In general, we see that under the second interpretation, the strategies Highest Predicted, Highest-Lowest Predicted, and Popularity still show their advantages over the baseline as DM techniques in most cases.

In the bottom row of Figure 1, we are observing again that the relative results of the strategies vary across CF models. For NeuMF, Popularity achieves the best model performance with any fixed amount of data, meaning the data collected by Popularity is the most informative in helping to promote the model's performance. This could be explained by the locality of the embeddings in NeuMF. Since Popularity queries mostly popular items, the embeddings of these items are quickly stabilized. This helps the model to achieve better performance in the test set, which is also dominated by the popular items. With LightGCN and MultiVAE, the advantage of Popularity is not so significant anymore, since

Table 1: Consumer-side unfairness metrics, defined using the user Gender attribute, and provider-side unfairness metrics of each model at steps 1, 3 and 5 of each AL strategy. For both the consumer-side and the provider-side unfairness metrics, lower values indicate better fairness. The best and second best metrics obtained at each step is **bolded** and underlined. For ΔNDCG and ΔPrecision, all the results are statistically significant under paired t-tests with $p < 0.05$, unless marked by *.

Model	AL Strategy	ΔNDCG			ΔPrecision			Statistical Parity			Equal Opportunity		
		t=1	t=3	t=5	t=1	t=3	t=5	t=1	t=3	t=5	t=1	t=3	t=5
NeuMF	**Highest Pred.**	0.026	0.049	0.060	0.023	0.044	0.054	0.615	**0.653**	**0.632**	**0.571**	**0.577**	**0.542**
	Hi-Lo Pred.	0.024	0.042	0.053	0.021	0.036	0.046	**0.614**	0.698	0.701	0.578	0.647	0.623
	Popularity	0.060	0.075	0.081	0.050	0.067	0.074	1.021	0.758	0.657	0.945	0.666	0.561
	Random	**0.011**	**0.019**	**0.031**	**0.011**	**0.013**	**0.021**	0.858	0.776	0.807	0.729	0.888	0.853
LightGCN	**Highest Pred.**	0.038	0.078	0.079	0.033	0.069	0.072	**0.651**	**0.729**	**0.673**	**0.622**	**0.668**	**0.594**
	Hi-Lo Pred.	0.032	0.065	0.076	0.030	0.056	0.068	0.676	0.838	0.790	0.643	0.763	0.698
	Popularity	0.059	0.075	0.085	0.051	0.068	0.076	1.048	0.917	0.820	0.946	0.747	0.682
	Random	**0.012**	**0.020**	**0.034**	**0.013**	**0.013***	**0.021**	0.720	0.776	0.798	0.692	0.874	0.847
MultiVAE	**Highest Pred.**	0.030	0.066	0.078	0.027	0.057	0.072	0.608	0.727	**0.677**	0.528	**0.625**	**0.581**
	Hi-Lo Pred.	0.025	0.050	0.072	0.022	0.042	0.061	**0.587**	0.738	0.767	**0.517**	0.664	0.676
	Popularity	0.062	0.082	0.084	0.049	0.075	0.080	1.006	0.772	0.705	0.939	0.664	0.593
	Random	**0.016**	**0.022***	**0.031**	**0.014**	**0.016***	**0.023**	0.648	**0.712**	0.766	0.606	0.810	0.807

these models depend either on the neighborhoods in the interaction graph or the entire interaction histories of users.

In summary, the results in Figure 1 demonstrate the effectiveness of the AL strategies over the baseline as DM techniques to attain better model performance under limited data collection, while also showcasing the need to choose AL strategy accordingly based on the CF model being used.

(Un)fairness Under AL Strategies (RQ2)

In this section we investigate the fairness implications of different AL strategies in deep CF models. In Table 1, we report the consumer-side unfairness metrics defined using the user Gender attribute and the provider-side unfairness metrics at steps $t = 1, 3, 5$ of each AL strategy. We focus on the first 5 steps here as this is the stage where all the metrics exhibit the most rapid changes. After this phase, the relative relationships between the four AL strategies remain stable for all the CF models, with the consumer-side unfairness metrics stay at similar values, and the provider-side unfairness metrics gradually decreasing for all the strategies. The consumer-side unfairness metrics defined using the user Age attribute are reported in Table 2, which shows similar results as the consumer-side unfairness metrics in Table 1, thus we will focus our discussion on Table 1 in this section.

We first note that for all the AL strategies, the consumer-side unfairness metrics are increased after each step of AL, suggesting the negative impact on consumer-side fairness of AL as a DM technique. As the system collects more feedback data from users, there is a greater amount of disparity in recommendation quality among users, no matter what AL strategy we use.

Comparing the AL strategies, we see that Random achieves the best consumer-side fairness, which is expected considering the fact that it solicits feedback data in a randomized manner and also achieves lower model performance. Highest-Lowest Predicted achieved the second lowest consumer-side unfairness, and Popularity demonstrates the highest level of unfairness. Comparing the results with those in Figure 1, there is a clear trade-off between model performance and consumer-side fairness, where the strategies that achieve better model performance are also leading to more degradation in consumer-side fairness.

For the provider-side unfairness metrics, Popularity generally starts with a very high value, and then gradually reduces to a value comparable with Highest Predicted and Highest-Lowest Predicted, showing the fact that Popularity suffers severely from popularity bias in the early stage of the AL process. Keeping in mind that the metrics are decreasing after the first 5 steps, we note that there is generally an increase-then-decrease trend for the other strategies. This shows that regardless of which strategy is used, the first few steps are highly biased towards the popular item categories, since they have a higher possibility of being included in the initial known set and being queried for in the early stage. Because the AL strategies do not query the same item twice (Equation 2), a higher variety of items will eventually be included in the known set, bringing the provider-side unfairness metrics down. When employing AL as a method of DM, this initial peak of provider-side unfairness should be taken into account of, especially if the number of AL steps is limited.

It is also worth noting that Highest Predicted achieves the best or second-best provider-side fairness in most of the cases. This could be explained by its property of being more tailored to each user's preferences and thus is less prone to popularity bias. Combining this with the results in Figure 1, Highest Predicted demonstrates its strength in balancing overall model performance and provider-side fairness. In the first two steps, Highest-Lowest Predicted also shows good provider-side fairness results. And although Popularity showed severe provider-side unfairness in the early stage, it is able to bring the unfairness metrics, especially REO@k, to a lower level, which could be due to its capability of soliciting more data in a fixed number of steps.

Table 2: Consumer-side unfairness metrics, defined using the user Age attribute, of each model at steps 1, 3 and 5 of each AL strategy. Lower values indicate better fairness. The best and second best metrics obtained at each step is **bolded** and underlined. All the results are statistically significant under paired t-tests with $p < 0.05$.

Model	AL Strat.	ΔNDCG			ΔPrecision		
		t=1	t=3	t=5	t=1	t=3	t=5
NeuMF	**High**	0.024	0.044	0.051	0.022	0.040	0.047
	Hi-Lo	0.023	0.040	0.046	0.021	0.035	0.040
	Pop	0.060	0.067	0.064	0.051	0.064	0.063
	Rand	**0.017**	**0.027**	**0.035**	**0.014**	**0.020**	**0.028**
LightGCN	**High**	0.033	0.074	0.083	0.031	0.064	0.074
	Hi-Lo	0.028	0.061	0.076	0.025	0.050	0.068
	Pop	0.065	0.075	0.078	0.054	0.069	0.071
	Rand	**0.016**	**0.025**	**0.040**	**0.014**	**0.017**	**0.028**
MultiVAE	**High**	0.022	0.053	0.066	0.021	0.048	0.063
	Hi-Lo	0.019	0.041	0.059	0.018	0.036	0.052
	Pop	0.061	0.074	0.074	0.052	0.069	0.071
	Rand	**0.017**	**0.026**	**0.037**	**0.015**	**0.020**	**0.029**

Overall, the results from Table 1, as well as Table 2, reveal the trade-off between model performance and consumer-side fairness. Furthermore, Highest Predicted provides the best balance between model performance and provider-side fairness, while all the strategies show an initial surge in provider-side unfairness.

Conclusion and Future Work

In this work, we analyze the implications of using DM techniques based on AL in RS. We first evaluate the impact of AL strategies as DM techniques on the performance of RS, investigating their behavior under two interpretations of minimizing data collection. We showcase the effectiveness of the AL strategies to promote model performance, while highlighting the importance of deploying a suitable AL strategy according to the CF model. These results demonstrate the usefulness of AL as DM techniques. Moreover, our implementation of DM in this work has other noteworthy advantages. For example, the conversion of ratings to implicit feedback further limits the amount of personal information. And a higher level of transparency is achieved by asking users for their consent to share feedback data.

We then investigate the fairness impacts of these strategies, demonstrating the trade-off between model performance and consumer-side fairness. The experimental results illustrate the capability of Highest Predicted to balance performance and provider-side fairness, while also showing the negative impact of all the AL strategies on provider-side fairness in the early stage.

As future work, we plan to extend our experiments to other AL strategies and consider different types of bias when applying DM techniques (e.g., popularity bias), to better understand their root causes in AL strategies. We also plan to conduct an online user study to compare AL strategies in real-world RS.

Acknowledgments

The authors acknowledge the Tufts University High Performance Compute Cluster (https://it.tufts.edu/high-performance-computing) which was utilized for the research reported in this paper. Author Sonboli's effort was supported by the CRA CIFellows program funded by the National Science Foundation under Grant No. 2127309. This research was supported by industry partners and the Research Council of Norway with funding to MediaFutures: Research Centre for Responsible Media Technology and Innovation, through the Centres for Research-based Innovation scheme, project number 309339.

References

Beheshti, A.; Ghodratnama, S.; Elahi, M.; and Farhood, H. 2022. *Social data analytics*. CRC press.

Biega, A. J.; Potash, P.; Daumé, H.; Diaz, F.; and Finck, M. 2020. Operationalizing the legal principle of data minimization for personalization. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, 399–408. New York, NY, USA: Association for Computing Machinery.

Bufi, S.; Paparella, V.; Anelli, V. W.; and Di Noia, T. 2025. Legal but unfair: Auditing the impact of data minimization on fairness and accuracy trade-off in recommender systems. In *Proceedings of the 33rd ACM Conference on User Modeling, Adaptation and Personalization*, 114–123.

Burke, R.; Adomavicius, G.; Bogers, T.; Di Noia, T.; Kowald, D.; Neidhardt, J.; Özgöbek, Ö.; Pera, M.; and Ziegler, J. 2024. Multistakeholder and multimethod evaluation. In *Evaluation Perspectives of Recommender Systems: Driving Research and Education (Dagstuhl Seminar 24211)*, 123–145. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik GmbH.

Chen, J.; Dong, H.; Wang, X.; Feng, F.; Wang, M.; and He, X. 2023. Bias and debias in recommender system: A survey and future directions. *ACM Trans. Inf. Syst.* 41(3).

Deldjoo, Y.; Jannach, D.; Bellogín, A.; Difonzo, A.; and Zanzonelli, D. 2024. Fairness in recommender systems: Research landscape and future directions. *User Modeling and User-Adapted Interaction* 34(1):59–108.

Elahi, M.; Repsys, V.; and Ricci, F. 2011. Rating elicitation strategies for collaborative filtering. In Huemer, C., and Setzer, T., eds., *E-Commerce and Web Technologies*, 160–171. Berlin, Heidelberg: Springer Berlin Heidelberg.

Elahi, M.; Ricci, F.; and Rubens, N. 2014. Active learning strategies for rating elicitation in collaborative filtering: A system-wide perspective. *ACM Trans. Intell. Syst. Technol.* 5(1).

Elahi, M.; Ricci, F.; and Rubens, N. 2016. A survey of active learning in collaborative filtering recommender systems. *Computer Science Review* 20:29–50.

Finck, M., and Biega, A. 2021. Reviving purpose limitation and data minimisation in personalisation, profiling and decision-making systems. In *Reviving Purpose Limitation and Data Minimisation in Personalisation, Profiling*

and Decision-Making Systems: Finck, Michèle— uBiega, Asia. [Sl]: SSRN.

Galdon Clavell, G.; Martín Zamorano, M.; Castillo, C.; Smith, O.; and Matic, A. 2020. Auditing algorithms: On lessons learned and the risks of data minimization. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, 265–271. New York, NY, USA: Association for Computing Machinery.

Golbandi, N.; Koren, Y.; and Lempel, R. 2010. On bootstrapping recommender systems. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, 1805–1808. New York, NY, USA: Association for Computing Machinery.

Harper, F. M., and Konstan, J. A. 2015. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.* 5(4).

He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T.-S. 2017. Neural collaborative filtering.

He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; and Wang, M. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation.

He, L.; Liu, N. N.; and Yang, Q. 2011. Active dual collaborative filtering with both item and attribute feedback. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, AAAI'11, 1186–1191. AAAI Press.

Karimi, R.; Freudenthaler, C.; Nanopoulos, A.; and Schmidt-Thieme, L. 2011. Non-myopic active learning for recommender systems based on matrix factorization. In *2011 IEEE International Conference on Information Reuse & Integration*, 299–303.

Liang, D.; Krishnan, R. G.; Hoffman, M. D.; and Jebara, T. 2018. Variational autoencoders for collaborative filtering.

Narayanan, A., and Shmatikov, V. 2008. Robust deanonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, 111–125.

Rashid, A.; Albert, I.; Cosley, D.; Lam, S.; McNee, S.; and Riedl, J. 2002. Getting to know you: Learning new user preferences in recommender systems. *International Conference on Intelligent User Interfaces, Proceedings IUI*.

Shanmugam, D.; Diaz, F.; Shabanian, S.; Finck, M.; and Biega, A. 2022. Learning to limit data collection via scaling laws: A computational interpretation for the legal principle of data minimization. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, 839–849. New York, NY, USA: Association for Computing Machinery.

Smith, J. J.; Madaio, M.; Burke, R.; and Fiesler, C. 2025. Pragmatic fairness: Evaluating ml fairness within the constraints of industry. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency (FAccT '25)*, 628–638. New York, NY, USA: Association for Computing Machinery.

Sonboli, N.; Burke, R.; Liu, Z.; and Mansoury, M. 2020. Fairness-aware recommendation with librec-auto. In *Fourteenth ACM Conference on Recommender Systems*, 594–596.

Sonboli, N.; Burke, R.; Ekstrand, M.; and Mehrotra, R. 2022. The multisided complexity of fairness in recommender systems. *AI Magazine* 43(2):164–176.

Sonboli, N.; Li, S.; Elahi, M.; and Biega, A. 2024. The trade-off between data minimization and fairness in collaborative filtering.

Wang, X.; He, X.; Wang, M.; Feng, F.; and Chua, T.-S. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '19, 165–174. ACM.

Wang, Y.; Ma, W.; Zhang, M.; Liu, Y.; and Ma, S. 2023. A survey on the fairness of recommender systems. *ACM Trans. Inf. Syst.* 41(3).

Yao, S., and Huang, B. 2017. Beyond parity: Fairness objectives for collaborative filtering.

Zhu, Z.; Wang, J.; and Caverlee, J. 2020. 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*, SIGIR '20, 449–458. New York, NY, USA: Association for Computing Machinery.