

Calibrated News Recommendation under Dynamic Feedback Loops: A Simulation Study

Caio Andrade

Mathematics and Computer Science Institute, University
of São Paulo
São Carlos, SP, Brazil
caio.tulio.andrade@usp.br

Marcelo Manzato

Mathematics and Computer Science Institute, University
of São Paulo
São Carlos, SP, Brazil
mmanzato@icmc.usp.br

ABSTRACT

Calibrated recommendations, a class of techniques designed to align recommendation lists with users' historical preference distribution, are largely underexplored under the dynamic feedback loop scenario. In this work, we use a dynamic recommendation simulation framework to study how calibration affects news recommendation over time, measuring miscalibration, diversity and fragmentation. We evaluate calibrated and uncalibrated recommendations on semi-synthetic simulation environments derived from Globo.com, a large-scale Brazilian news portal dataset. Our results show that calibration consistently increases fragmentation over time, while slightly reducing catalog coverage. Filtering out previously interacted items (exposure filtering), a common practice in recommender systems, also plays a role: calibrated recommendations start out more diverse, but diversity decreases over time as top-ranked articles get filtered out, while uncalibrated recommendations become more diverse overall, despite cyclical fluctuations. The same pattern holds for miscalibration, which increases over time for calibrated variants but decreases for uncalibrated ones.

KEYWORDS

Recommender Systems, Calibration, Feedback Loop, News Recommendation, User Simulation, Fairness

1 INTRODUCTION

Recommender Systems (RSs) help users navigate large online catalogs by learning from past interactions and from the behavior of similar users. The goal of such systems is to alleviate decision fatigue and deliver relevant content to the user needs [48]. Particularly, in News Recommendation Systems (NRS) a difficult balance is needed: users should receive relevant content according to their interests or political views, however, over personalization can limit users exposure towards different news or viewpoints [46].

This delicate balance can be further challenged by biases present in the interaction data or model design, which may amplify the overexposure of popular news while under-representing niche articles. This issue is particularly concerning in dynamic recommendation settings, where users consume recommended news, these interactions are logged, and future recommendations are trained from this very feedback, creating a loop that can amplify existing biases. Over

time, this cycle may deepen skewed exposure patterns, contributing to filter bubbles, rabbit holes, and preference drift. [33, 41, 42].

One strategy that limits this drift is calibrated recommendations [52]. Steck proposes that recommendations should approximate the distribution of genres or categories observed in users' past interactions, so that recommendation lists better reflect users' diverse interests. Calibration has become an active research topic in recent years, although it remains a relatively specific area within RS fairness [16]. Recent work has begun to incorporate preference evolution into calibrated recommendation, for example by selecting more representative slices of user history, down-weighting older interactions, or questioning the use of stale preferences as calibration targets [17, 36, 57]. Moving towards dynamic evaluation is fundamental in a scenario where the long-term role of recommender systems is called into question: whether they should empower users to discover content aligned with their genuine interests, or whether, as purely accuracy-driven offline evaluation often overlooks, they risk narrowing the online experience over time by reinforcing existing consumption patterns and distorting user preferences [24, 25, 30].

To the best of our knowledge, the role of calibration remains underexplored in dynamic recommendation scenarios, where users consume recommendations, those interactions are logged and used to train the recommender model, and the cycle repeats. Discussions involving calibration with NRS, especially in the dynamic context, are also scarce. Even though previous studies explain the role that calibration plays on NRS, no work has been proposed to do so in a dynamic scenario, connecting its effects to other diversity-related metrics [54]: if we take into account model retraining, exposure filtering (that is, avoiding recommending previously interacted items, a common practice with recommender systems) and user interaction in the feedback loop, how does news diversity change over time? Would users be exposed to similar articles, triggering common topics for discussion and enabling healthy conversations about current events, or would they be isolated into bubbles? Moreover, does calibration tip the delicate balance between personalization and common exposure between users? In summary, we hope to address the following research questions, distilled from the discussion above:

- (1) **RQ1:** How does calibration affect miscalibration under dynamic feedback loops?
- (2) **RQ2:** How does calibration affect diversity and catalog coverage over time?
- (3) **RQ3:** Does calibration increase news fragmentation in dynamic recommendation scenarios?

- (4) **RQ4:** How does the previously-seen item filter interact with calibration over retraining cycles?

In this paper, we address those questions by studying the effects of calibrated recommendation in a simulated dynamic recommendation scenario. We apply calibration in a feedback loop scenario based on the Globo.com dataset, a large-scale Brazilian news portal [21]. We measure average miscalibration and other fairness-related metrics, such as catalog coverage, intra-list similarity and news fragmentation [54]. Our simulation approach, detailed in Section 4.1, allows us to emulate the interaction between user, model and data and measure how news recommendations evolves through time with or without calibration.

The remainder of this paper is structured as follows. Section 2 covers technical background on calibrated recommendations, Section 3 covers related works on calibration, fairness, and dynamic recommender systems. Section 4 details our user simulation, including preference evolution and acquisition. We describe our experiments in Section 5, discuss results in Section 6, and conclude in Section 7. The code used for the simulation¹, the recommendation model implementation², and calibrated recommendations³ are publicly available, as well as the experimental results.

2 BACKGROUND

Recommender systems have traditionally aimed to produce recommendations that are faithful to users' preferences, initially with a strong focus on predictive accuracy [48]. As the field matured, research increasingly considered broader effects of recommendation, including popularity bias, exposure unfairness, diversity loss, and unequal utility across user groups [3, 20, 26]. Within this broader fairness landscape, calibrated recommendation was introduced by Steck [52] to ensure that recommendation lists reflect the distribution of interests observed in users' historical interactions.

Calibration is a subfield of fairness in recommender systems that focuses on whether a recommendation list reflects the multiple interests of an individual user [1, 28]. Following Steck [52], the user's historical profile is represented as a distribution $p(g|u)$ over genres or categories, computed from the user's past interactions. A recommendation list $R(u)$ induces a corresponding distribution $q(g|u)$.

In practice, calibration is commonly implemented as a re-ranking procedure based on Maximal Marginal Relevance (MMR) [9], where the final list balances relevance and calibration:

$$R_{\text{calibrated}}(u) = \arg \max_{\mathcal{L} \subseteq C(u), |\mathcal{L}|=N} [(1 - \lambda)S(\mathcal{L}) - \lambda \text{Dist}(p(\cdot|u), q(\cdot|u, \mathcal{L}))]. \quad (1)$$

Here, $C(u)$ is the candidate set, $\mathcal{L}$ is the candidate calibrated recommendation list, $S(\mathcal{L})$ is its relevance score, and λ controls the trade-off between relevance and calibration. $\mathcal{L}$ is greedily assembled as items from $C(u)$ that maximize Equation 1 are chosen. the higher λ gets, the more calibrated the list becomes, at the cost of relevance. This value is typically either hard-coded to a fixed high value, or obtained through more sophisticated approaches, such as

hyperparameter tuning or analyzing the variance of genres in the user profile [18, 51].

The degree to which a list is calibrated is typically measured by Miscalibration (MC), introduced by Abdollahpour et al [2]. The metric is defined as the average distance between the user's historical preference distribution and the distribution induced by the recommendation list:

$$MC(p, q) = \frac{1}{|U|} \sum_{u \in U} \text{Dist}(p(\cdot|u), q(\cdot|u)), \quad (2)$$

where the distance function $\text{Dist}(p(\cdot|u), q(\cdot|u))$ is typically KL divergence [34] or Hellinger distance. A list is considered well calibrated when $q(g|u) \approx p(g|u)$.

3 RELATED WORK

A growing body of research has investigated the relationship between calibration and other fairness-related phenomena. One prominent line of work examines the connection between popularity bias and calibrated recommendation. While some studies analyze how calibration influences popularity bias, and vice versa [2, 29], others explicitly leverage users' preferences for popular or niche items during calibration [49] or introduce calibration directly into model learning through in-processing approaches, such as adapting BPR [47] to account for users' preference toward popular or niche items [19]. In addition, Lin et al. [37] investigate how different aspects of a user profile, such as taste diversity and intensity, may contribute to miscalibration. Other works connect calibration to recommendation diversity, either by discussing their shared goals [28] or by showing that calibration may reduce overall diversity [16].

3.0.1 Calibration and News Recommender Systems. Calibration has been largely underexplored in the news recommendation domain in comparison to others, such as movies or music [16]: Treuillier et al. [53] explore a fairness framework for NRS that jointly optimizes fairness and diversity, using a calibration-like approach to build the users' profile; in a subsequent work, the authors critique standard diversity evaluation approaches in NRS, arguing that high diversity in recommendations does not systematically lead to more diverse news consumption, and call for temporal and personalized evaluation frameworks [53]; finally, Vrijenhoek et al. [54] connects beyond-accuracy diversity metrics, such as Calibration and Fragmentation, to normative models of democracy and the role news recommenders should play in each.

3.0.2 Preference Evolution and Calibration. Another recent line of work has questioned the assumption, present in Steck's original formulation and subsequent publications, that users' preferences are static over time. Zhao et al. criticize standard calibration for enforcing that the distribution of recommended items should match users' past interests, without accounting for taste evolution. The authors refer to this mismatch as taste distortion, which occurs when calibrated recommendations fail to keep up with the user's evolving preferences.

In a similar direction, but with a more exploratory focus, Lin et al. [36] propose a calibration approach in which the user's history is partitioned into timestamp-based segments. Miscalibration is then measured between the user's current preference profile and each

¹https://github.com/caiotda/dynamicTasteDistortion/tree/webmedia_paper_globo

²<https://github.com/caiotda/bpr-mf>

³<https://github.com/caiotda/calibrated-recommendations/tree/main>

segment, and the segment with the lowest miscalibration is selected as the most representative of the user's current preferences. This segment is subsequently used as the reference profile for calibrating the recommendation list.

More recently, Da Silva et al. [17] proposed modifications to Steck's original formulation to account for diverse and evolving preferences. In particular, they down-weight older interactions when calibrating a list by applying a timestamp-based $w_{u,i}$ in $p(g|u)$. As a result, preference evolution is explicitly incorporated into the calibration process. However, these works share a common limitation: they evaluate calibration and diversity in static recommendation scenarios.

3.0.3 Literature Gap. While the recommender systems community has long acknowledged that preferences shift over time [31], this issue has only recently received attention in the context of calibration. Calibration in NRS (and recommender systems more broadly) has been studied predominantly in static settings, rather than in dynamic scenarios where users and recommendation models continuously influence one another [10, 39, 43, 45]. Previous studies have examined how feedback loops, a core aspect of dynamic recommender systems, relate to filter bubbles [5], diversity, and popularity bias [38, 58]. Additionally, a concurrent extended abstract shares our motivation of exploring calibration in dynamic recommendation settings [35]. While their work focuses on different aspects such as the λ trade-off parameter between accuracy and calibration and comparing distribution distance measures, our work explores different calibration strategies, a distinct user simulation model, and the effects of user propensity towards acquiring new preferences. As of the writing of this paper, their full results have not yet been made public.

To the best of our knowledge, the behavior of calibration under dynamic feedback loops, where user preferences, recommendation exposure, and model retraining continuously influence one another, remains largely unexplored. In this work, we address this gap through a user simulation framework that allows us to study how calibrated recommendations shape and are shaped by evolving user preferences over time.

4 METHODOLOGY

One of the main challenges in studying dynamic recommendation is modeling the feedback loop, which can be investigated either through longitudinal user studies on online recommendation platforms or through simulation [8, 38, 56, 58]. Due to the high cost and complexity of conducting long-term studies with real users, we focus on simulation. Our goal is to provide an initial step toward understanding calibration in dynamic settings and to motivate future longitudinal studies with real users, similarly to prior work on user perception of calibration in static scenarios [4]. A high-level overview of the experimental setup is shown in Figure 1 and is further explained in the following subsections.

4.1 User Simulation

Modeling user behavior faithfully is itself a challenging task [44]. Our modeling assumptions, which extend previous work on user simulation and modeling [11, 58], are as follows:

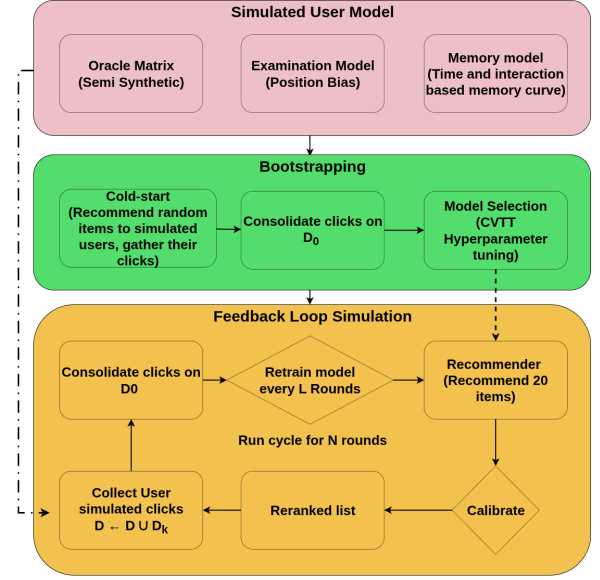


Figure 1: Overview of the simulation framework. The framework consists of three stages: simulated user modeling, bootstrapping, and feedback loop simulation. Dotted lines indicate component reuse across stages. L denotes the retraining frequency, N the number of simulation rounds, and $\mathcal{D}$ the accumulated set of simulated clicks. The simulated user Model is further explained in Section 4.1, the bootstrapping step and feedback loop in Sections 4.2, 5.3 and 5.4.

- (1) Users are biased toward selecting top-ranked items. We model click-position bias, a well-documented phenomenon in recommender systems [12, 14].
- (2) User clicks are independent within a session: interacting with the j -th recommended item does not affect the probability of clicking on the $(j + 1)$ -th item.
- (3) User preferences are affected by recommended items, modeled through a fixed probability α that a user acquires interest in an item they examined but did not click.
- (4) Repeated interactions are not allowed: once a user interacts with an item, it is not recommended again before retraining.
- (5) User behavior is learned from a base dataset, following a semi-synthetic approach to derive preferences and interaction frequency from observed data.

We operationalize these assumptions through three components: an examination model and an oracle model, based on prior work on simulated recommendation [38, 58], which determine which items are clicked; and a memory model, which accounts for preference acquisition and forgetting, drawing on human learning and forgetting curves [11, 13, 40]. To introduce temporal dynamics, we estimate the average time between consecutive ratings and use it to simulate interaction timestamps.

4.1.1 Generating interaction timestamps. Because we assume that interactions are independent, we model the time between consecutive interactions as a random variable following an exponential

distribution. For each user u , let $T(u) = \{t_0, t_1, \dots, t_m\}$ be the ordered set of observed interaction timestamps. We compute the set of observed inter-arrival times per user, $\Delta T(u) = \{t_k - t_{k-1} \mid k = 1, \dots, m\}$, and use their median as the scale parameter of the exponential distribution:

$$\beta_u = \text{median}(\Delta T(u)). \quad (3)$$

Therefore, the time between two consecutive simulated interactions for user u is sampled as:

$$\Delta t_l(u) \sim \text{Exponential}(\beta_u), \quad (4)$$

The timestamp of the l -th simulation round for user u is then defined as:

$$t_l = t_{l-1} + \Delta t_l(u), \quad (5)$$

where $t_0 = 0$. Since samples from the exponential distribution are strictly positive, we have $t_l > t_{l-1}$ for every simulation round l .

4.1.2 Oracle model. To define the relevance of a given user-item pair, we follow a semi-synthetic approach frequently employed in simulation-based studies [5, 39, 58]. We train a predictive model on the observed interactions from Globo.com dataset and use it to fill in the missing entries of the user-item matrix. We refer to this model as the oracle model.

At the end of this process, we obtain a dense binary relevance matrix M , referred to as the oracle matrix, where $M_{u,i} = 1$ if item i is considered relevant to user u , and $M_{u,i} = 0$ otherwise. Details on oracle model selection are provided in Section 5.2.

4.1.3 Examination Model. Let $\mathcal{I} = \{i_1, i_2, \dots, i_{|\mathcal{I}|}\}$ be the set of items, and let $R(u) = \{i_1, i_2, \dots, i_k\}$ be a recommendation list of size k produced for user u , where $i_j \in \mathcal{I}$ denotes the item at position j . We model user examination using a position-biased examination model, since users are more likely to inspect items displayed near the top of a recommendation list. For each position j , we define the examination probability, following Joachims et al., as:

$$P_{\text{exam}}(j) = \frac{1}{\log_2(j+1)}. \quad (6)$$

This probability decreases monotonically with the item position, reflecting the tendency of users to examine top-ranked items more frequently than lower-ranked ones [12, 14].

To simulate whether user u examines the item at position j , we draw a random variable $z_j \sim \text{Uniform}(0, 1)$ and define the examination indicator as:

$$o_j = 1[z_j \leq P_{\text{exam}}(j)], \quad (7)$$

where $o_j = 1$ denotes that the item was examined, and $o_j = 0$ denotes that it was not examined. Since z_j is uniformly distributed, the probability that $o_j = 1$ is exactly $P_{\text{exam}}(j)$. Moreover, because each z_j is drawn independently, examinations across positions are mutually independent.

4.1.4 Combining the Examination and Oracle Models. To simulate the interaction between user u and a recommended item i_j , where i_j denotes the item at position j in the recommendation list, we combine the examination and oracle models. Let $c(u, i_j)$ be a binary variable indicating whether user u clicked item i_j :

$$c(u, i_j) = o_j \cdot M_{u,i_j}, \quad (8)$$

where o_j indicates whether the item at position j was examined, and M_{u,i_j} indicates whether item i_j is relevant to user u according to the oracle matrix. Thus, $c(u, i_j) = 1$ only when the item was both examined and relevant to the user.

4.1.5 Memory Model. To understand how the feedback loop affects users' interests, we include mechanisms for both preference acquisition and forgetting. To simulate preference forgetting, we draw inspiration from Ebbinghaus's forgetting curve [40] and from prior work on user modeling in recommender systems [11, 13, 15].

Following these works, we define the probability that user u forgets a preference for genre g at time t as the complement of a retention function:

$$P_{\text{forget}}(g|t, u) = 1 - \exp(-f(t, u, g)), \quad (9)$$

where $f(t, u, g)$ is a non-negative function that increases with time, so that retention strictly decreases the longer a preference goes unreinforced. Following Chen et al., we compute retention for user u toward genre g as a function of both elapsed time and how frequently the user has interacted with that genre:

$$f(t, u, g) = 1 + \left(\frac{t - t_{\min}(u)}{t_{\max}(u) - t_{\min}(u)} - \frac{n(u, g)}{\sum_{g' \in \mathcal{G}} n(u, g')} \right), \quad (10)$$

where $t_{\min}(u)$ and $t_{\max}(u)$ are the earliest and latest interaction timestamps of user u . The first term min-max normalizes t , making decay scale-invariant to absolute time; the second is the fraction of user u 's interactions belonging to genre g . Both terms are bounded to $[0, 1]$, and we add 1 to keep $f(t, u, g) \in [0, 2]$ non-negative, so that users with stronger affinity toward a genre are less likely to forget preferences related to it.

We apply forgetting at the article level, assigning each article a_j the forgetting probability of its genre: $P_{\text{forget}}(a_j|t, u) := P_{\text{forget}}(g(a_j)|t, u)$. This is applied to every relevant article recommended but not clicked by the user — i.e., for each $a_j \in R(u)$ such that $M_{u,a_j} = 1$ and $c(u, a_j) = 0$, the user's preference for a_j may be forgotten according to $P_{\text{forget}}(a_j|t, u)$.

Preference acquisition is modeled through a hyperparameter α , representing the probability that a user acquires interest in an article they examined but that was initially irrelevant according to the oracle model. Formally, for each item $i_j \in R(u)$ such that $M_{u,i_j} = 0$ and $o_j = 1$, we update the oracle matrix as:

$$M'_{u,i_j} = \begin{cases} 1, & \text{if } z_j \sim \text{Uniform}(0, 1) \leq \alpha, \\ M_{u,i_j}, & \text{otherwise.} \end{cases} \quad (11)$$

where M' is the updated oracle matrix and z_j is defined in Equation 7. Lower values of α model more conservative users, while higher values model users more prone to acquiring new preferences through exposure. We hardcode $\alpha = 0.1$ and leave exploration of different acquisition levels for future work.

4.2 Dynamic Recommendation simulation

The feedback loop is a core component of dynamic recommendation [12, 45]: users interact with recommended items, these interactions are logged, and the model is retrained or updated using this data to generate future recommendations. This cycle allows the system to adapt over time, but it may also reinforce biases and shape user preferences through repeated exposure.

4.2.1 Cold-starting the Simulation. We first create an initial cold-start dataset, denoted as D_0 , containing clicks generated by the simulated users. To bootstrap the simulation, we use a random recommendation policy: each user receives 20 randomly selected items, and the user model described in Section 4.1 determines which items are clicked. The resulting clicks are added to D_0 .

This process is repeated until D_0 contains approximately 500 thousand positive interactions, which provides a sufficient amount of data to train the initial recommender model.

4.2.2 Dynamic Recommendation. As the recommendation model, we use a BPR-based matrix factorization model [47], due to its simplicity, applicability to implicit feedback scenarios, and efficient optimization through negative sampling. Since our user simulation explicitly includes position bias, as described in Section 4.1, we modify the original BPR loss by adding a position-based weight that reflects the position at which an item was clicked. Therefore, lower ranked clicks are considered more important as they are removed from top-rank click biases.

Given a set $\mathcal{D}$ of training tuples (u, i, j) , where i is a positive item and j is a sampled negative item for user u , we define the loss as:

$$\mathcal{L}_{\text{BPR-pos}} = \frac{1}{|\mathcal{D}|} \sum_{(u,i,j) \in \mathcal{D}} \frac{1}{\log_2(1 + k_{u,i})} \cdot (-\log \sigma(\hat{y}_{u,i} - \hat{y}_{u,j})) + \gamma_{\text{reg}} \|\theta\|^2, \quad (12)$$

where $\hat{y}_{u,i}$ and $\hat{y}_{u,j}$ are the predicted scores for the positive and negative items, respectively, γ_{reg} is the regularization hyperparameter, and $\|\theta\|^2$ is the squared norm of the model parameters. To debias click position, we set a click-aware term, where $k_{u,i}$ is the position at which positive item i was clicked by user u .

After cold-starting the simulation, we train the recommender model on the bootstrapped click set D_0 . We then perform hyperparameter tuning on this bootstrapped set, partitioning it into training, validation, and test splits using cross-validation through time (CVTT). Model performance is evaluated using Mean Average Precision at 5 (MAP@5), a standard ranking metric for binary feedback data. We choose a relatively short cut-off point due to how we model user examination behavior, which concentrates interactions in the first few positions of a ranking. A recommended item i is considered a hit for user u when $c(u, i) = 1$.

At each simulation round ℓ , the recommender model generates recommendation lists for the simulated users. The clicked items are collected into a new interaction set D_ℓ , which is then added to the accumulated training data:

$$D \leftarrow D \cup D_\ell. \quad (13)$$

Every L rounds, the model is retrained using the updated interaction set D .

At each round, we also measure taste distortion by comparing the genre distribution of the interactions collected at round ℓ with the genre distribution of the initial bootstrapped set D_0 . We use average miscalibration based on Hellinger distance to quantify how user preferences change over time. We choose Hellinger Distance instead of other probability distribution metrics, such as KL divergence, due to being bounded in a $[0, 1]$ interval and being less sensitive

by small fluctuations, something that must be taken into account due to the random components in our simulation, such as the click model detailed in section 4.1.3. Besides a calibration metric, we also measure diversity related metrics, such as Intra-list diversity (ILD), catalog coverage and news fragmentation.

We run three recommendation policies: a calibrated recommendation policy, which applies Steck's classical formulation of calibrated recommendation, assigning equal importance to all interactions in the user's history [52]; an uncalibrated BPR policy, which uses the raw output of the recommendation model without any re-ranking; and a random recommendation policy, included as a baseline to isolate the effects of user interaction patterns from those of model personalization. We refrain from testing different calibration formulations, such as time-based, with adaptive weighting or popularity-bias-aware calibration [17, 18, 49] and leave this to future work.

5 EXPERIMENTAL SETTING

Our experiments consist of three main steps: oracle model generation, click bootstrapping, and dynamic simulation of user-item interactions. Each step is described below before presenting the experimental results.

5.1 Datasets and Preprocessing

The simulated users in our experiments are derived from Globo.com⁴ one of the largest and most relevant news outlets in Brazil. Since running dynamic simulations over time is computationally expensive, we focus on a dataset that is manageable in size while still containing the information required for our study. In particular, because we are interested in preference evolution and calibration, we require a news dataset that provides interaction timestamps, item genre or category information, and impression logs, all of which are available in the Globo.com dataset. Others datasets, such as MovieLens [22], Yelp [6], and MIND [55] fit this criteria. We choose Globo.com specifically for its relevance to the Brazilian context, the density of its interaction logs, and its focus on news recommendation. With that said, our experimental setup imposes no domain limitations beyond the user-item information described above.

The dataset undergoes a filtering step to ensure sufficient signal for training. We remove articles with no associated category, as well as articles belonging to categories with fewer than 36 interactions (the 50th percentile), and users with fewer than 10 interactions. This reduces the dataset from 322,897 users, 46,033 items, 316 genres and a sparsity of 0.9998, to 17,749 users, 14,853 items, 130 genres and a sparsity of 0.995.

5.2 Oracle model selection

To evaluate matrix-filling performance, we test three factorization models as oracle models: SVD, SVD++, and NMF [32]. Hyperparameter selection follows a cross-validation through time (CVTT) approach, which we use instead of user-based or random partitioning in order to preserve the temporal structure of preference

⁴<https://www.kaggle.com/datasets/gspmoreira/news-portal-user-interactions-by-globocom>

evolution. The data are split chronologically into training, validation, and out-of-time (OOT) test sets, using a 70/15/15 split. The validation set is used for hyperparameter tuning, while the OOT test set is used to measure final model performance. The hyperparameter grids used for model selection are available in the extended material ⁵. We use the same grid for SVD and SVD++. This step was implemented using the Surprise library [23].

We select the best oracle model based on F1 score on the OOT test set, since our goal is to identify the model that best predicts users' future binary preferences, which aligns with our experimental design. Because evaluating all hyperparameter combinations would be computationally costly, we perform random search cross-validation [7], sampling 20 random configurations per model and per dataset.

Because Globo.com dataset contains only implicit feedback signals, we perform random-negative sampling for oracle model fitting and evaluation so that we have positives and negatives, sampling 3 negatives per positive. We report the best-performing variants for each model and their F1 scores in Table 1. The full results are provided in the paper's github repository to keep the main text concise.

Table 1: Best hyperparameter configurations per model and dataset

Model	f	e	η	γ	F1
SVD++	15	10	0.01	0.02	0.658
SVD	15	10	0.01	0.02	0.658
NMF	15	100	—	$\gamma_u=0.05, \gamma_i=0.02$	0.634

e : number of epochs, f : number of factors, η : learning rate; γ : regularization; γ_i : per user/item regularization. F1 reports Test F1.

Finally, we generate a fully filled binary relevance matrix M that stores the relevance of each user-item pair. For efficiency reasons, we randomly select 1,000 users for the simulation and store their inferred preferences, since storing the complete dense matrix for all users and items is computationally impractical. Although the simulation is restricted to these 1,000 sampled users, the oracle models are trained on the full user-item interaction matrix. Removing the remaining users before training would weaken the collaborative signal available to the factorization models, degrade the learned user-item structure, and ultimately reduce prediction quality for the sampled users.

5.3 Bootstrapped Clicks

Using the user model described in Section 4.1, we generate an initial dataset of user clicks. This process consists of randomly recommending 20 items to each user, collecting the resulting clicks, and repeating the procedure until approximately 500,000 positive interactions are recorded. Although this dataset size is arbitrary, it provides a reasonable number of positive interactions to train the initial BPR model. We also prevent repeated recommendations by masking items that the user has previously interacted with.

Even though items are recommended randomly at this stage, simulated users only click items that are relevant according to the

oracle matrix, with a higher probability of examining top-ranked positions. Therefore, the bootstrapped set provides a proxy for the initial preference distribution for the simulated users, which we denote as H_0 , before their interactions are influenced by a recommender system and its feedback loop. In Section 6, we report miscalibration by comparing the recommendations or interactions produced at each simulation round against this initial preference distribution.

The bootstrapped set derived from Globo.com comprises 1,000 users and 2,944 items spanning 123 genres, with approximately 500,000 interactions and a sparsity of 0.83.

5.4 Dynamic Recommendation Model Selection

To select the BPR hyperparameters used in the simulation, we perform CVTT on the bootstrapped clicks rather than the base datasets, as simulated users behave differently from real users, and tuning on the base data would create a mismatch between model selection and the simulation setting.

The model is optimized to predict clicked items. Since clicks are generated by both the oracle model and the stochastic examination model, this task reflects the noisy implicit-feedback setting used in our simulation and resembles a production environment with noisy implicit feedback, making this a challenging task. We therefore use cold-start MAP as the model selection criterion, while refraining from reporting accuracy-oriented metrics in the main experiments, since our focus is calibration and diversity related metrics. The hyperparameters tested via CVTT are available in the extended material ⁶, and the validation results are reported in Section 5.3.

To verify that the model is capable of learning meaningful preferences when sufficient signal is available, we additionally evaluate the best-performing hyperparameters selected on the bootstrapped set derived from Globo.com. The best configuration uses a learning rate of 0.0001, regularization of 0.001, 5 negative samples, and 7 epochs, yielding a MAP@5 of 0.13, which is a competitive result for BPR [47].

5.5 Simulation

In our experiments, we compare three variants: an uncalibrated BPR model with the position-weighted loss function described in Equation 12; one calibrated variants built on top of this model implementing Steck's calibration with constant weights and a random recommendation policy, which serves as a baseline.

For each experiment, we measure the following metrics at every simulation round, all computed @10, as this defines the size of the list over which calibration is calculated and ensures comparability between calibrated and uncalibrated variants:

- (1) **Miscalibration**: measured using Hellinger distance. We compare the distortion between what's being recommended and the initial set of bootstrapped preferences.
- (2) **Diversity**: measured as the complement of the average pairwise Jaccard similarity between item categories within a recommendation list.
- (3) **Catalog coverage**: measured as the proportion of distinct items recommended across all users in each round, capturing

⁵<https://anonymous.4open.science/r/dynamicTasteDistortion-webmedia>

⁶<https://anonymous.4open.science/r/dynamicTasteDistortion-webmedia/dynamicTasteDistortion>

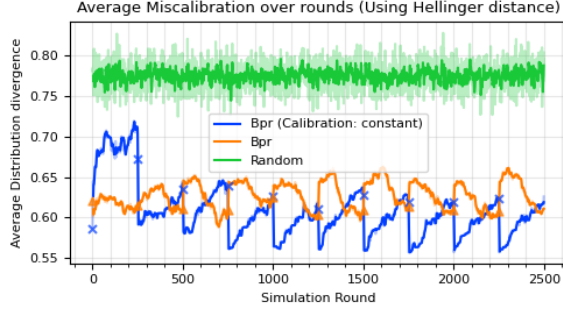


Figure 2: Miscalibration over simulation rounds, comparing personalized approaches (with and without calibration) against an unpersonalized one.

the system’s ability to expose the broader catalog rather than concentrating recommendations on a narrow subset of items.

- (4) **Fragmentation:** derived from the Average Rank-Biased Overlap of item ids between users. Essentially, the higher the fragmentation, the higher the chance that two users are being delivered non-overlapping news recommendations. We base our implementation on Vrijenhoek et al.’s definitions and refer to the source paper for further discussions [54].

We run the experiments for $N = 2,500$ rounds, retraining the model every $L = 250$ rounds, and avoid recommending items the user interacted with in the last L rounds, a common practice in recommender systems. These values let us observe how recommendations evolve over time under sustained user interaction, as well as the impact of retraining. Due to the high computational cost of the simulation, ablation studies on these hyperparameters are left for future work. Finally, experiments are run for 10 repetitions, and we report the average metric per simulation.

6 RESULTS

Figure 2 reports Miscalibration through time, between random recommendation, BPR and Calibrated BPR and each highlighted point represents a model retraining. We can see that the personalized variants undergo a “warmup” phase until the first retraining at round 250, with their behavior stabilizing afterward. Even though the calibrated variant consistently achieves a lower miscalibration score than the uncalibrated version, miscalibration decreases over time for the uncalibrated version. This is likely due to an inherent model bias, which starts out recommending biased items and ends up with more calibrated ones over time. For the calibrated variant, however, miscalibration increases over time. A plausible explanation is the previously-seen item filter, which progressively removes well-calibrated items from the candidate set. We leave a direct ablation of this mechanism for future work. We even see that calibration ends up performing the same or worse than the uncalibrated version over time, before retraining resets this behavior. This suggests that model and concept drift can be strong factors that introduce miscalibration.

The same trend can be observed for recommendation diversity in Figure 3, where retraining events again act as inflection points for both variants. After retraining, the uncalibrated recommendations

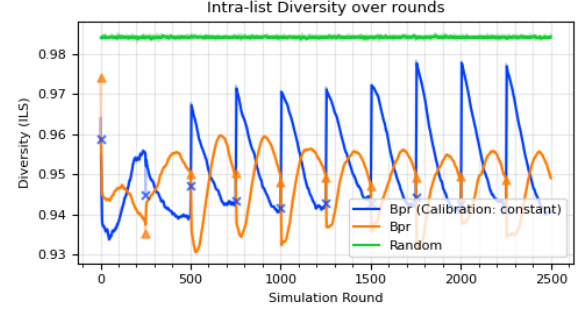


Figure 3: Diversity over simulation rounds. Diversity is measured as the complement of ILS.

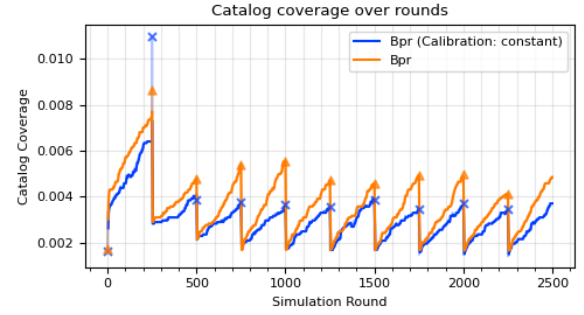


Figure 4: Catalog coverage over rounds (random recommendation omitted for clarity).

get less diverse, likely due to concentration on a smaller subset of news categories, as users interact with them over successive rounds, and recommendations start getting more diverse. Interestingly, these recommendations exhibit an almost cyclical behavior after reaching a higher level of diversity. This might be due to how we model preference evolution over time, but this would need to be further evaluated by ablation studies on how users are simulated, which we leave for future work. On the other hand, recommendation diversity decreases over time for the calibrated variant. The same policy of not recommending repeated items drives this trend: as users interact with the top positions of the calibrated list, diversity diminishes over time. Because Globo.com has over 100 genres, each user’s preference is calibrated to a very fine-grained level. This means that even small shifts in user interactions can push calibrated recommendations toward an increasingly narrow slice of the catalog, accelerating diversity loss.

Even though diversity changes differently over time for each variant, catalog coverage always increases over time, as we can see in Figure 4. We see that calibration consistently achieves lower catalog coverage values in comparison to the uncalibrated one, which aligns with previous findings on static calibration. As we retrain the model, we reset the previously seen items, which triggers a drop in catalog coverage. The model then recommends a small subset of items, with calibration diminishing the catalog even further.

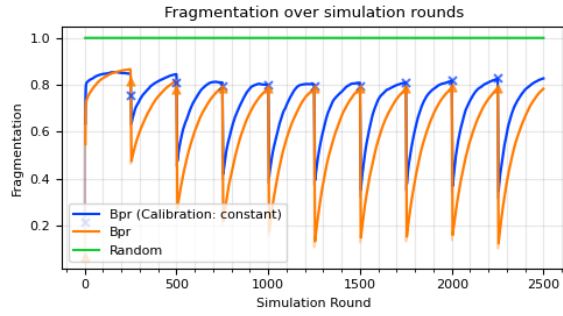


Figure 5: Fragmentation over simulation rounds

Finally, we observe fragmentation over time in Figure 5. Consistent with the behavior in terms of catalog coverage, we observe that every time we reset the previously seen filter, we observe a drop in fragmentation, delivering more personalized news per user while decreasing isolation. As users interact with the most relevant news, fragmentation starts to go up, regardless of the re-ranking strategy. Calibration consistently shows higher degrees of fragmentation compared to uncalibrated variants. It’s interesting to note that fragmentation consistently increases over time, which might be an indicative of a subset of items dominating the top-ranked positions of recommendations at the start of the cycle - yielding smaller fragmentation scores - and eventually yielding more diverse results, resulting in a higher fragmentation score.

6.1 Discussion

Our results suggest that, given enough interactions, uncalibrated recommendations eventually converge towards calibrated recommendations in terms of miscalibration, as top-ranked items get interacted with and lower-ranked ones are delivered. This highlights the role of previously-seen item filtering in shaping long-term miscalibration, as avoiding repeated recommendations forces the system to explore a wider range of content over time.

Calibration also introduces an interesting effect on diversity and fragmentation over time. Consistent with previous studies on news recommendation [54], calibration increases fragmentation for Globo.com alongside achieving a lower miscalibration score than the uncalibrated version, which is expected since calibration delivers highly personalized articles to each user, reducing the common sphere of news shared across them.

Whether this is desirable depends on the democratic role a news recommender is meant to play [54]. Under a Liberal model, high fragmentation is acceptable, as recommendations should reflect individual preferences and autonomy; under Participatory or Deliberative models, lower fragmentation is preferred, as these value a shared public sphere with a common set of issues and topics. The increased fragmentation introduced by calibration may thus be a feature or a flaw depending on the platform’s editorial mission.

Furthermore, calibration increases diversity over time, as ensuring different genres are represented in the recommendation list acts as a form of diversification. We leave to future work whether this effect is dataset-specific, given that Globo.com comprises over 100 genres with each item assigned to a single genre.

7 CONCLUSION

Our results show that the exposure filter is a core driver of metric variation in a feedback loop scenario, capable of overruling calibration’s own effects: it increases miscalibration over time for calibrated recommendations while decreasing it for uncalibrated ones, with diversity following the same inverted pattern. Calibration itself, however, consistently increases fragmentation over time, a potential concern for NRS that aim to sustain a shared set of articles and topics for debate, while specializing recommendations further toward users’ past preferences, reflected in lower catalog coverage and miscalibration at the start of each retrain cycle.

Future work should investigate the use of other models as recommenders, such as neighborhood-based and, specially, sequential models. We highlight the latter as news are hardly consumed in isolation, but rather as a string of related articles, sometimes with converging political viewpoints or diverging ones. Other works could also explore the impact of calibration on news polarization in Dynamic NRS, either via simulation or via real users: even though we observe an increase in fragmentation, due to our dataset limitation, we cannot conclude whether those fragmented communities are a manifestation of filter bubbles or rabbit holes [41, 42].

One limitation of our study is the lack of ablation studies on oracle components introduced in Section 4; This is due to the high computational cost of running simulation rounds over time for different variants. The effect of different user behavior profiles (such as being more biased towards popular items, instead of top-ranked ones) and simulation approaches on the effects of calibration are left as future work, alongside understanding the effect of different recommendation models, such as sequential and neural models.

Finally, offline evaluation, while necessary, remains an imperfect proxy for real user experience, specially with NRS. Simulation offers a partial bridge by modeling temporal dynamics and preference drift that static metrics cannot capture, but it is not a substitute for real-world validation. A longitudinal user study would be an important next step to assess whether calibration strategies in dynamic settings translate into tangible improvements in user satisfaction and engagement. Such studies remain scarce in recent literature, where offline benchmarks continue to dominate as the primary form of evaluation [50].

8 ACKNOWLEDGMENTS

The authors acknowledge the financial support from FAPESP, CAPES, CNPq, and INCT-TILD-IAR. During manuscript preparation, the authors used Claude (Anthropic) to assist in rewriting existing text for formatting and cohesion. All content was reviewed and verified by the authors, who take full responsibility for the final manuscript.

ETHICAL CONSIDERATIONS

This study uses only publicly available datasets (e.g. globo.com dataset) and did not involve direct interaction with human participants. Only the information provided in the public datasets was used, with no attempt to identify or re-identify users. Therefore, the study is consistent with Brazilian Resolution CNS No. 510/2016 and did not require evaluation by a Research Ethics Committee.

REFERENCES

- [1] Himan Abdollahpour, Masoud Mansoury, and Robin Burke. 2020. The Connection Between Popularity Bias, Calibration, and Fairness in Recommendation. In *Proceedings of the 14th ACM Conference on Recommender Systems* (Virtual Event, Brazil) (RecSys '20). Association for Computing Machinery, New York, NY, USA, 726–731. doi:10.1145/3383313.3418487
- [2] Himan Abdollahpour, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. 2019. The impact of popularity bias on fairness and calibration in recommendation. *arXiv preprint arXiv:1910.05755* (2019).
- [3] Himan Abdollahpour, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. 2019. The unfairness of popularity bias in recommendation. *arXiv preprint arXiv:1907.13286* (2019).
- [4] Gabrielle Alves, Dietmar Jannach, Rodrigo Ferrari De Souza, and Marcelo Garcia Manzato. 2024. User Perception of Fairness-Calibrated Recommendations. In *Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization* (Cagliari, Italy) (UMAP '24). Association for Computing Machinery, New York, NY, USA, 78–88. doi:10.1145/3627043.3659558
- [5] Guy Aridor, Duarte Goncalves, and Shan Sikdar. 2020. Deconstructing the filter bubble: User decision-making and recommender systems. In *Proceedings of the 14th ACM conference on recommender systems*. 82–91.
- [6] Nabihha Asghar. 2016. Yelp dataset challenge: Review rating prediction. *arXiv preprint arXiv:1605.05362* (2016).
- [7] James Bergstra and Yoshua Bengio. 2012. Random search for hyper-parameter optimization. *Journal of machine learning research* 13, 2 (2012).
- [8] Dimitrios Bountouridis, Jaron Harambam, Mykola Makhortyk, Mónica Marrero, Nava Tintarev, and Claudia Hauff. 2019. SIREN: A Simulation Framework for Understanding the Effects of Recommender Systems in Online News Environments. In *Proceedings of the Conference on Fairness, Accountability, and Transparency* (Atlanta, GA, USA) (FAT* '19). Association for Computing Machinery, New York, NY, USA, 150–159. doi:10.1145/3287560.3287583
- [9] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. 335–336.
- [10] Micah D Carroll, Anca Dragan, Stuart Russell, and Dylan Hadfield-Menell. 2022. Estimating and penalizing induced preference shifts in recommender systems. In *International Conference on Machine Learning*. PMLR, 2686–2708.
- [11] Hongqi Chen, Zhiyong Feng, Shizhan Chen, Hongyue Wu, Yingchao Sun, Jingyu Li, Qinghang Gao, Lu Zhang, and Xiao Xue. 2025. Incorporating Forgetting Curve and Memory Replay for Evolving Socially-aware Recommendation. *Information Processing & Management* 62, 3 (2025), 104070. doi:10.1016/j.ipm.2025.104070
- [12] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and Debias in Recommender System: A Survey and Future Directions. *ACM Trans. Inf. Syst.* 41, 3, Article 67 (Feb. 2023), 39 pages. doi:10.1145/3564284
- [13] Jun Chen, Chaokun Wang, and Jianmin Wang. 2014. Modeling the Interest-Forgetting Curve for Music Recommendation. In *Proceedings of the 22nd ACM International Conference on Multimedia* (Orlando, Florida, USA) (MM '14). Association for Computing Machinery, New York, NY, USA, 921–924. doi:10.1145/2647868.2654954
- [14] Andrew Collins, Dominika Tkaczyk, Akiko Aizawa, and Joeran Beel. 2018. A study of position bias in digital library recommender systems.
- [15] Zhihong Cui, Xiangguo Sun, Hongxu Chen, Li Pan, Li zhen Cui, Shijun Liu, and Guandong Xu. 2024. Dynamic Recommendation Based on Graph Diffusion and Ebbinghaus Curve. *IEEE Transactions on Computational Social Systems* 11 (2024), 2755–2764. <https://api.semanticscholar.org/CorpusID:258448820>
- [16] Diego da Silva and Dietmar Jannach. 2026. Calibrated Recommendations: Survey and Future Directions. *ACM Trans. Recomm. Syst.* 4, 3, Article 43 (March 2026), 32 pages. doi:10.1145/3789266
- [17] Diego Corrêa da Silva, Dietmar Jannach, and Frederico Araújo Durão. 2025. Considering Time and Feature Entropy in Calibrated Recommendations. *ACM Transactions on Intelligent Systems and Technology* 16, 3 (2025), 1–29.
- [18] Diego Corrêa da Silva, Marcelo Garcia Manzato, and Frederico Araújo Durão. 2021. Exploiting personalized calibration and metrics for fairness recommendation. *Expert Systems with Applications* 181 (2021), 115112. doi:10.1016/j.eswa.2021.115112
- [19] Rodrigo Ferrari de Souza and Marcelo Garcia Manzato. 2024. Uma abordagem em etapa de processamento para redução do viés de popularidade. In *Brazilian Symposium on Multimedia and the Web (WebMedia)*. SBC, 310–317.
- [20] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogin, Alessandro Difonzo, and Dario Zanzonelli. 2024. Fairness in recommender systems: research landscape and future directions. *User Modeling and User-Adapted Interaction* 34, 1 (2024), 59–108.
- [21] P Moreira Gabriel De Souza, Dietmar Jannach, and Adilson Marques Da Cunha. 2019. Contextual hybrid session-based news recommendation with recurrent neural networks. *IEEE access* 7 (2019), 169185–169203.
- [22] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4, Article 19 (Dec. 2015), 19 pages. doi:10.1145/2827872
- [23] Nicolas Hug. 2020. Surprise: A Python library for recommender systems. *Journal of Open Source Software* 5, 52 (2020), 2174.
- [24] Dietmar Jannach and Gediminas Adomavicius. 2016. Recommendations with a purpose. In *Proceedings of the 10th ACM conference on recommender systems*. 7–10.
- [25] Dietmar Jannach, Alan Said, Marko Tkalcić, and Markus Zanker. 2025. Recommender systems for good (RS4Good): survey of use cases and a call to action for research that matters. *ACM Transactions on Recommender Systems* (2025).
- [26] Dietmar Jannach and Markus Zanker. 2012. Value and impact of recommender systems. In *Recommender systems handbook*. Springer, 519–546.
- [27] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, Filip Radlinski, and Geri Gay. 2007. Evaluating the accuracy of implicit feedback from clicks and query reformulations in Web search. *ACM Trans. Inf. Syst.* 25, 2 (April 2007), 7–es. doi:10.1145/1229179.1229181
- [28] Mesut Kaya and Derek Bridge. 2019. A comparison of calibrated and intent-aware recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems* (Copenhagen, Denmark) (RecSys '19). Association for Computing Machinery, New York, NY, USA, 151–159. doi:10.1145/3298689.3347045
- [29] Anastasiia Klimashevskaya, Mehdi Elahi, Dietmar Jannach, Christoph Trattner, and Lars Skjærven. 2022. Mitigating popularity bias in recommendation: potential and limits of calibration approaches. In *International Workshop on Algorithmic Bias in Search and Recommendation*. Springer, 82–90.
- [30] Bart P. Knijnenburg, Saadhika Sivakumar, and Darcia Wilkinson. 2016. Recommender Systems for Self-Actualization. In *Proceedings of the 10th ACM Conference on Recommender Systems* (Boston, Massachusetts, USA) (RecSys '16). Association for Computing Machinery, New York, NY, USA, 11–14. doi:10.1145/2959100.2959189
- [31] Yehuda Koren. 2010. Collaborative filtering with temporal dynamics. *Commun. ACM* 53, 4 (April 2010), 89–97. doi:10.1145/1721654.1721677
- [32] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix Factorization Techniques for Recommender Systems. *Computer* 42, 8 (Aug. 2009), 30–37. doi:10.1109/MC.2009.263
- [33] Joshua Krook and Jan Blockx. 2023. Recommender systems, autonomy and user engagement. In *Proceedings of the first international symposium on trustworthy autonomous systems*. 1–9.
- [34] S. Kullback and R. A. Leibler. 1951. On Information and Sufficiency. *The Annals of Mathematical Statistics* 22, 1 (1951), 79–86. doi:10.1214/aoms/1177729694
- [35] Kun Lin. 2026. Calibration in Dynamic Recommender Systems. In *Proceedings of the Nineteenth ACM International Conference on Web Search and Data Mining* (USA) (WSDM '26). Association for Computing Machinery, New York, NY, USA, 1296–1298. doi:10.1145/3773966.3784983
- [36] Kun Lin, Masoud Mansoury, Farzad Eskandarian, Milad Sabouri, and Bamshad Mobasher. 2024. Beyond Static Calibration: The Impact of User Preference Dynamics on Calibrated Recommendation. In *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization*. 86–91.
- [37] Kun Lin, Nasim Sonboli, Bamshad Mobasher, and Robin Burke. 2020. Calibration in Collaborative Filtering Recommender Systems: a User-Centered Analysis. In *Proceedings of the 31st ACM Conference on Hypertext and Social Media* (Virtual Event, USA) (HT '20). Association for Computing Machinery, New York, NY, USA, 197–206. doi:10.1145/3372923.3404793
- [38] Masoud Mansoury, Himan Abdollahpour, Mykola Pechenizkiy, Bamshad Mobasher, and Robin Burke. 2020. Feedback Loop and Bias Amplification in Recommender Systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management* (Virtual Event, Ireland) (CIKM '20). Association for Computing Machinery, New York, NY, USA, 2145–2148. doi:10.1145/3340531.3412152
- [39] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. 2020. Controlling fairness and bias in dynamic learning-to-rank. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 429–438.
- [40] Jaap MJ Murre and Joeri Dros. 2015. Replication and analysis of Ebbinghaus' forgetting curve. *PloS one* 10, 7 (2015), e0120644.
- [41] Derek O'Callaghan, Derek Greene, Maura Conway, Joe Carthy, and Pádraig Cunningham. 2015. Down the (white) rabbit hole: The extreme right and online recommender systems. *Social Science Computer Review* 33, 4 (2015), 459–478.
- [42] Eli Pariser. 2011. *The filter bubble: What the Internet is hiding from you*. penguin UK.
- [43] Bruno Laporais Pereira. 2017. *Recomendação online de músicas usando feedback implícito*. In "Masters dissertation". Universidade Federal de Minas Gerais.
- [44] Erasmo Purificato, Ludovico Boratto, and Ernesto William De Luca. 2024. User modeling and user profiling: A comprehensive survey. *arXiv preprint arXiv:2402.09660* (2024).
- [45] Chhavi Rana and Sanjay Jain. 2012. A study of the dynamic features of recommender systems. *Artificial Intelligence Review* 43 (11 2012). doi:10.1007/s10462-012-9359-6
- [46] Shaina Raza and Chen Ding. 2022. News recommender system: a review of recent progress, challenges, and opportunities. *Artificial Intelligence Review* 55, 1 (2022), 749–800.

- [47] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).
- [48] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2010. Introduction to recommender systems handbook. In *Recommender systems handbook*. Springer, 1–35.
- [49] Andre Sacilotti, Rodrigo Ferrari de Souza, and Marcelo Garcia Manzato. 2023. Counteracting popularity-bias and improving diversity through calibrated recommendations. In *Proceedings of the 25th International Conference on Enterprise Information Systems (ICEIS)*.
- [50] Alan Said, Maria Soledad Pera, and Michael D Ekstrand. 2025. We're Still Doing It (All) Wrong: Recommender Systems, Fifteen Years Later. *arXiv preprint arXiv:2509.09414* (2025).
- [51] Diego Silva and Frederico Durão. 2022. Explorando Justiça em Sistemas de Recomendação. In *Anais Estendidos do XXVIII Simpósio Brasileiro de Sistemas Multimídia e Web* (Curitiba). SBC, Porto Alegre, RS, Brasil, 11–14. doi:10.5753/webmedia_estendido.2022.225303
- [52] Harald Steck. 2018. Calibrated recommendations. In *Proceedings of the 12th ACM conference on recommender systems*. 154–162.
- [53] Céline Treuillier, Sylvain Castagnos, Özlem Özgöbek, and Armelle Brun. 2024. Beyond Trade-offs: Unveiling Fairness-Constrained Diversity in News Recommender Systems. In *Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization* (Cagliari, Italy) (UMAP '24). Association for Computing Machinery, New York, NY, USA, 143–148. doi:10.1145/3627043.3659571
- [54] Sanne Vrijenhoek, Mesut Kaya, Nadia Metoui, Judith Möller, Daan Odijk, and Natali Helberger. 2021. Recommenders with a mission: assessing diversity in news recommendations. In *Proceedings of the 2021 conference on human information interaction and retrieval*. 173–183.
- [55] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. 2020. Mind: A large-scale dataset for news recommendation. In *Proceedings of the 58th annual meeting of the association for computational linguistics*. 3597–3606.
- [56] Sirui Yao, Yoni Halpern, Nithum Thain, Xuezhi Wang, Kang Lee, Flavien Prost, Ed Chi, Jilin Chen, and Alex Beutel. 2021. Measuring Recommender System Effects with Simulated Users. doi:10.48550/arXiv.2101.04526
- [57] Xing Zhao, Ziwei Zhu, and James Caverlee. 2021. Rabbit holes and taste distortion: Distribution-aware recommendation with evolving interests. In *Proceedings of the Web Conference 2021*. 888–899.
- [58] Ziwei Zhu, Yun He, Xing Zhao, and James Caverlee. 2021. Popularity Bias in Dynamic Recommendation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* (Virtual Event, Singapore) (KDD '21). Association for Computing Machinery, New York, NY, USA, 2439–2449. doi:10.1145/3447548.3467376