Correcting Exposure Bias for Link Recommendation

Shantanu Gupta[1 2] **Hao Wang**[3] **Zachary Lipton**[2] **Yuyang Wang**[4]

Abstract

Link prediction methods are frequently applied in recommender systems, e.g., to suggest citations for academic papers or friends in social networks. However, exposure bias can arise when users are systematically underexposed to certain relevant items. For example, in citation networks, authors might be more likely to encounter papers from their own field and thus cite them preferentially. This bias can propagate through naively trained link predictors, leading to both biased evaluation and high generalization error (as assessed by true relevance). Moreover, this bias can be exacerbated by feedback loops. We propose estimators that leverage known exposure probabilities to mitigate this bias and consequent feedback loops. Next, we provide a loss function for *learning* the exposure probabilities from data. Finally, experiments on semi-synthetic data based on real-world citation networks, show that our methods reliably identify (truly) relevant citations. Additionally, our methods lead to greater diversity in the recommended papers' fields of study. The code is available at github.com/shantanu95/exposure-bias-link-rec.

1. Introduction

Diverse application domains, including both citation networks and social networks, are characterized by graph-structured data. Here, nodes represent entities (like papers or users) and edges represent associations between two nodes (like citations, friendships, or follows). Link recommender systems (RSs) leverage node attributes and existing links to suggest new nodes that a given node *should* link to (Li et al., 2017; Bai et al., 2019; Ma et al., 2020). Typically, RSs are trained and evaluated directly on the observed graph, raising concerns about exposure bias—many missing links are false negatives, and did not form due to lack of exposure rather than a lack of affinity.

Consider the example of an RS that recommends relevant citations to authors given attributes of their paper (like title, abstract, etc.). In this case, equally relevant papers from different fields of study (FOS) might be less cited historically because authors have been preferentially exposed to papers in their own FOS. In the *observed* citation graph, a number of relevant papers are observed as *not cited* because the user was not exposed to those papers. Thus evaluating a link RS directly on the observed graph may yield a biased estimate of the true risk.

Exposure bias can exacerbate popularity bias, causing relevant but unpopular items to not be shown (Chen et al., 2020). In social networks, diverse recommendations can help users form links with communities they would otherwise not discover (Li et al., 2017; Brandão et al., 2013). In citation networks, exposure bias can also lead to lines of research being duplicated across fields. Examples include *model-based science* and *linear canonical transforms*, which were developed in isolation (Vincenot, 2018; Liberman & Wolf, 2015). Thus it would be valuable to have an RS that recommends relevant low-exposure nodes.

In this paper, we call the probability that a node is exposed to another node the *propensity score*; and we call the probability that, given exposure, a node links to another node the *link probability*. An RS trained directly on the observed data will underestimate the link probability for low propensity nodes relative to high propensity nodes. We demonstrate this with a simple example in the context of academic citation recommendation.

Example 1 (**Exposure Bias**). *Let's say that there are two FOS: Machine Learning (ML) and Physics (PH), with* n *papers in each. An ML researcher is looking for papers to cite. The probability of them being exposed to papers in ML and PH is* 0.9 *and* 0.6*, respectively. Given exposure, the probabilities that they cite papers from ML and PH are* 0.8 *and* 0.8*, respectively. In the observed data, we will see* $\approx 0.72n (= 0.9 \times 0.8n)$ *ML papers cited and* $\approx 0.48n (= 0.6 \times 0.8n)$ *PH papers cited. Thus, if we directly learn link probabilities from the observed data, the probability of citing a PH paper will be underestimated (*0.48 *instead*

[1] Work done while interning at Amazon [2] Machine Learning Department, Carnegie Mellon University [3] Department of Computer Science, Rutgers University [4] Amazon Web Services (AWS) AI Labs, Palo Alto, CA, USA. Correspondence to: Shantanu Gupta <shantang@cs.cmu.edu>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

of 0.8*) more than that of an ML paper (*0.72 *instead of* 0.8*). This shows that equally relevant papers with lower propensity may be deemed less relevant.*

To begin, we show that evaluating an RS naively on the observed data provides a misleading measure of its risk. Instead, we argue that an RS should be evaluated via the risk that would have been incurred had every user been exposed to every node. We call this the *true risk*. We propose three estimators of the true risk that use known propensity scores for estimation (Section 3). The key idea is to weight the positive and negative links using functions of the propensity scores and link probabilities. Each of the three estimators uses a different weighting scheme. We provide sufficient conditions for when they will have lower bias than the naive estimator for the true risk. We then derive a generalization bound that shows that, with high probability, the true risk is close to the risk estimated by our proposed methods. We use this bound to motivate a loss function that can be used to simultaneously *learn* the link probabilities and propensity scores (Section 4). Next, under a simplified model of link recommendation, where nodes belong to one of a finite number of categories, we prove that feedback loops arise under exposure bias and that they worsen at a faster rate for lower propensity nodes (Section 5). We further show that accounting for exposure bias can help alleviate them.

We empirically validate our methods on real-world citation data from the Microsoft Academic Graph (MAG) (Sinha et al., 2015) (Section 6). Since true exposure values are not available in the real data, we construct a semi-synthetic data with simulated exposure and link probabilities. Our methods lead to higher precision and recall against true citations than the naive method. On real data, our methods maintain comparable performance to the naive method on metrics computed against the observed data and recommend more papers from different fields-of-study.

2. Related Work

There is a rich literature for correcting bias in RS. Swaminathan & Joachims (2015) present a counterfactual risk minimization framework for learning from logged bandit feedback. Joachims et al. (2017) use a counterfactual inference framework to counteract selection bias in click data. Schnabel et al. (2016) propose unbiased performance estimators for RS that use known propensity scores when explicit item ratings are observed with selection bias. Ma & Chen (2019) recover propensities under the low nuclear norm assumption. Wang et al. (2020b) use exposure data to construct a substitute for unobserved confounders. Wang et al. (2021) show that bandit algorithms can lead to an unfair allocation of exposure across arms, and to overcome this issue, they propose an alternative formulation, where each arm receives exposure proportional to its merit. The implicit feedback setting, where user interactions, such as clicking and listening (as opposed to explicit ratings), are used to train the RS, is more closely related to our setting. It is known that in this setting, some negative examples are false negatives due to exposure bias (Jeunen, 2019, Section 4.1). Yang et al. (2018) use inverse propensity scoring to create an unbiased evaluator for this setting using inverse-propensity-scoring based methods. Liang et al. (2016b) model exposure as a latent variable and incorporate it into a collaborative-filtering approach. Liang et al. (2016a) use exposure and click models to re-weight samples to make unbiased predictions. Our work leverages ideas from these works, especially the approach of re-weighting samples to counter the bias. However, this work addresses the item recommendation regime and the methods do not translate to the link prediction setting.

Chang & Blei (2009) develop a relational topic model for link prediction in document graphs. Wang et al. (2017) extend this work by incorporating deep learning under the framework of Bayesian deep learning (Wang et al., 2015; Wang & Yeung, 2016; 2020). In social networks, learning-based methods and proximity-based methods are leveraged (Wang & Li, 2013; Li et al., 2017). Masrour et al. (2020) study filter bubbles in link prediction and propose a method to recommend more diverse links. Citation recommenders use paper data and metadata for training (Beel et al., 2016; Ma et al., 2020). Some systems use local citation contexts to improve predictions (Wang et al., 2020a; Haruna et al., 2017). In contrast, our goal in this work is to augment existing models such that they account for exposure bias during both training and evaluation.

Addressing feedback loops in RSs, Chaney et al. (2018) and Mansoury et al. (2020) use simulations to demonstrate that they can arise, amplifying popularity bias and user homogeneity. Sun et al. (2019) present several matrix-factorization-based debiasing algorithms to prevent feedback loops. Sinha et al. (2016) propose a method to identify the items affected by feedback loops and recover the user's intrinsic preferences. Jiang et al. (2019) show that feedback loops can create echo-chambers and filter bubbles. Zhao et al. (2017) show that models amplify biases in training data and propose a constraint-based method to mitigate this. In contemporaneous work, Wang & Russakovsky (2021) extend this work and propose another metric for measuring bias and empirically show that it disentangles the direction of amplification.

3. Estimating Risk under Exposure Bias

Notation. Our dataset is a directed graph $\mathcal{G}(V, E)$, where $V = \{v_1, \dots, v_n\}$ is the set of n nodes and E is the set of edges, s.t. $(i, j) \in E$ denotes a link from v_i to v_j. We denote by $\mathcal{U} = \{(i, j) : i \in [n], j \in [n], \text{s.t.}\, i \neq j\}$ the possible (including missing) links in the graph; by π_{ij} the *propensity*, i.e., the probability that v_i is exposed to v_j; and by y_{ij} the *link probability*, i.e., the probability that v_i links to v_j *conditional* on exposure to v_j. The binary random variable o'_{ij} represents if v_i links to v_j assuming exposure to v_j; the binary random variable a_{ij} represents if v_i is exposed to v_j; and the binary random variable o_{ij} representing if v_i links to v_j. Thus the data generating process for $\mathcal{G}(V, E)$ is as follows: $\forall (i,j) \in \mathcal{U}$, we have

$$\begin{aligned} o'_{ij} &\sim \text{Ber}(y_{ij}), \\ a_{ij} &\sim \text{Ber}(\pi_{ij}), \\ o_{ij} &= o'_{ij} a_{ij}, \end{aligned}$$

where Ber(.) is the Bernoulli distribution. The predicted link probability is $\widehat{y}_{ij}$ and the estimated propensity is $\widehat{\pi}_{ij}$. The predicted link outcome is $\widehat{o}_{ij} = \mathbb{1}(\widehat{y}_{ij} \geq 0.5)$. As an example, consider a citation graph. Here, each v_i is an academic paper, π_{ij} is the probability that authors of v_i are exposed to v_j, and y_{ij} is the probability that v_i cites v_j conditional on exposure to v_j.

Definition 1 (True Risk). *This is the risk of the predictions $\widehat{y}$ on the graph that would have been generated if all nodes were exposed to all other nodes, i.e., if $\forall\ (i,j) \in \mathcal{U},\ \pi_{ij} = 1$. The true risk is defined as*

$$\begin{aligned} R(\widehat{y}) &= \mathbf{E}_{o'}\left[\frac{1}{|\mathcal{U}|}\sum_{(i,j)\in\mathcal{U}} \delta(o'_{ij}, \widehat{y}_{ij})\right] \\ &= \frac{1}{|\mathcal{U}|}\sum_{(i,j)\in\mathcal{U}} [y_{ij}\delta(1, \widehat{y}_{ij}) + (1 - y_{ij})\delta(0, \widehat{y}_{ij})], \end{aligned}$$

where δ is some loss function (for example, log-loss).

True risk is different from the risk of the predictions on the observed graph as some relevant links are missing due to a lack of exposure. Thus the performance of an RS should be evaluated based on the true risk since it correctly accounts for relevant but low-exposure nodes.

In order to compare the biases and variances of the estimators we propose, we make Assumption 1 in this section. All proofs for this section are in Appendix A.

Assumption 1. *The loss function δ satisfies the following:*

1. *It only depends on the predicted binary outcome, i.e., $\delta(o_{ij}, \widehat{y}_{ij}) = \delta(o_{ij}, \widehat{o}_{ij})$,*
2. $\delta(0, 0) = \delta(1, 1) = 0$, *and*
3. $\delta(0, 1) = \delta(1, 0) := \Delta$.

Naive Estimator. One approach to estimating the true risk is to directly use the observed graph. We call this the naive estimator. It is defined as

$$\widehat{R}_{\text{naive}}(\widehat{y}) = \frac{1}{|\mathcal{U}|}\sum_{(i,j)\in\mathcal{U}} \delta(o_{ij}, \widehat{o}_{ij}).$$

Lemma 3.1. *The bias and variance of $\widehat{R}_{naive}(\widehat{o})$ are*

$$\begin{aligned} B(\widehat{R}_{naive}) &= \left|\mathbf{E}[\widehat{R}_{naive}] - R(\widehat{o})\right| \\ &= \frac{\Delta}{|\mathcal{U}|}\left|\sum_{(i,j)\in\mathcal{U}} y_{ij}(1 - \pi_{ij})(1 - 2\widehat{o}_{ij})\right|, \\ Var(\widehat{R}_{naive}) &= \frac{\Delta^2}{|\mathcal{U}|^2}\sum_{(i,j)\in\mathcal{U}} y_{ij}\pi_{ij}(1 - y_{ij}\pi_{ij}). \end{aligned}$$

Lemma 3.1 shows that $\widehat{R}_{\text{naive}}$ is a biased estimator of the true risk. $\widehat{R}_{\text{naive}}$ will be unbiased only if either all nodes are exposed to all the others, i.e., if $\forall (i,j) \in \mathcal{U},\ \pi_{ij} = 1$, or if all nodes are irrelevant to all the others, i.e., if $\forall (i,j) \in \mathcal{U},\ y_{ij} = 0$. Thus evaluating an RS using $\widehat{R}_{\text{naive}}$ can be misleading. We propose three estimators that leverage learned propensities $\widehat{\pi}$ and link probabilities $\widehat{y}$ to weight the examples to correct for this bias.

Estimator $\widehat{R}_w$. The first estimator we propose is

$$\widehat{R}_w(\widehat{y}, \widehat{\pi}) = \frac{1}{|\mathcal{U}|}\sum_{(i,j)\in\mathcal{U}} w_{ij}\delta(o_{ij}, \widehat{o}_{ij}), \text{ where} \quad (1)$$

$$w_{ij} = \frac{o_{ij}}{\widehat{\pi}_{ij}} + (1 - o_{ij})\psi_{ij},\ \psi_{ij} = \frac{1 - \widehat{y}_{ij}}{1 - \widehat{\pi}_{ij}\widehat{y}_{ij}}. \quad (2)$$

In $\widehat{R}_w$, the positive examples are up-weighted according to the inverse propensity. The negative examples are down-weighted (as $\psi_{ij} \leq 1$). Intuitively, this weighting corrects for the fact that, in the observed graph, some positive examples are observed as negative examples since the nodes are exposed according to the propensities π.

Lemma 3.2. *The bias and variance of $\widehat{R}_w$ are*

$$\begin{aligned} B(\widehat{R}_w) = \frac{\Delta}{|\mathcal{U}|}\Bigg|\sum_{(i,j)\in\mathcal{U}} \Bigg[&(1 - \widehat{o}_{ij})y_{ij}\left(1 - \frac{\pi_{ij}}{\widehat{\pi}_{ij}}\right) + \\ &\widehat{o}_{ij}\left(1 - y_{ij} - (1 - y_{ij}\pi_{ij})\psi_{ij}\right)\Bigg]\Bigg|, \quad (3) \end{aligned}$$

$$Var(\widehat{R}_w) = \frac{\Delta^2}{|\mathcal{U}|^2}\sum_{(i,j)\in\mathcal{U}} y_{ij}\pi_{ij}(1 - y_{ij}\pi_{ij})v_{ij},$$

where $v_{ij} = \dfrac{1 - \widehat{o}_{ij}}{\widehat{\pi}^2_{ij}} + \widehat{o}_{ij}\psi^2_{ij}$.

Lemma 3.2 shows that $\widehat{R}_w$ will be unbiased if the propensities and link probabilities are estimated correctly, i.e., if $\forall (i,j) \in \mathcal{U},\ \widehat{\pi}_{ij} = \pi_{ij}$ and $\widehat{y}_{ij} = y_{ij}$. We later derive sufficient conditions for when $\widehat{R}_w$ will have lower bias than $\widehat{R}_{\text{naive}}$ even if π and y are incorrectly estimated.

Estimator $\widehat{R}_{\text{PU}}$. We adapt an unbiased estimator proposed by Bekker et al. (2019) for the positive-and-unlabeled (PU) setting. The idea is to remove an appropriate number of negative examples for each positive example. We have

$$\widehat{R}_{\text{PU}}(\widehat{y},\widehat{\pi}) = \frac{1}{|\mathcal{U}|}\sum_{(i,j)\in\mathcal{U}}\left[w_{ij}\delta(o_{ij},\widehat{o}_{ij}) + w'_{ij}\delta(0,\widehat{o}_{ij})\right],$$

$$\text{where } w_{ij} = \frac{o_{ij}}{\widehat{\pi}_{ij}} + (1-o_{ij}),\ w'_{ij} = o_{ij}\left(1-\frac{1}{\widehat{\pi}_{ij}}\right).$$

We weight the positive examples by the inverse propensity and for each positive example, remove a negative example weighted by $|w'_{ij}|$.

Lemma 3.3. *The bias and variance of $\widehat{R}_{PU}$ are*

$$B(\widehat{R}_{PU}) = \frac{\Delta}{|\mathcal{U}|}\left|\sum_{(i,j)\in\mathcal{U}} y_{ij}\left(1-\frac{\pi_{ij}}{\widehat{\pi}_{ij}}\right)(1-2\widehat{o}_{ij})\right|,$$

$$Var(\widehat{R}_{PU}) = \frac{\Delta^2}{|\mathcal{U}|^2}\sum_{(i,j)\in\mathcal{U}}\frac{y_{ij}\pi_{ij}(1-y_{ij}\pi_{ij})}{\widehat{\pi}^2_{ij}}.$$

$\widehat{R}_{\text{PU}}$ will be unbiased when $\forall(i,j)\in\mathcal{U},\ \widehat{\pi}_{ij}=\pi_{ij}$.

Estimator $\widehat{R}_{\text{AP}}$. $\widehat{R}_{\text{AP}}$ adds positive examples for each negative example. It is defined as

$$\widehat{R}_{\text{AP}}(\widehat{y},\widehat{\pi}) = \frac{1}{|\mathcal{U}|}\sum_{(i,j)\in\mathcal{U}}\left[w_{ij}\delta(o_{ij},\widehat{o}_{ij}) + w'_{ij}\delta(1,\widehat{o}_{ij})\right],$$

$$\text{where } w_{ij} = o_{ij} + (1-o_{ij})\psi_{ij}, w'_{ij} = (1-o_{ij})\tau_{ij},$$

$$\tau_{ij} = \left(\frac{\widehat{y}_{ij}(1-\widehat{\pi}_{ij})}{1-\widehat{\pi}_{ij}\widehat{y}_{ij}}\right).$$

Lemma 3.4. *The bias and variance of $\widehat{R}_{AP}$ are*

$$B(\widehat{R}_{AP}) = \frac{\Delta}{|\mathcal{U}|}\Bigg|\sum_{(i,j)\in\mathcal{U}}(1-\widehat{o}_{ij})[(1-\pi_{ij})y_{ij} - (1-\pi_{ij}y_{ij})\tau_{ij}] + \widehat{o}_{ij}\left(1-y_{ij}-(1-y_{ij}\pi_{ij})\psi_{ij}\right)\Bigg|,$$

$$Var(\widehat{R}_{AP}) = \frac{\Delta^2}{|\mathcal{U}|^2}\sum_{(i,j)\in\mathcal{U}} y_{ij}\pi_{ij}(1-y_{ij}\pi_{ij})\psi^2_{ij},$$

where ψ is defined in Eq. 2.

$\widehat{R}_{\text{AP}}$ is unbiased if $\forall(i,j)\in\mathcal{U},\ \widehat{\pi}_{ij}=\pi_{ij}$ and $\widehat{y}_{ij}=y_{ij}$.

Theorem 3.1 (Comparison of Variances). *For all values of $\widehat{\pi},\widehat{y}$, we have $Var(\widehat{R}_{AP}) < Var(\widehat{R}_{naive})$, and $Var(\widehat{R}_{AP}) < Var(\widehat{R}_{w}) < Var(\widehat{R}_{PU})$.*

In order to compare the biases, we make the following simplifying assumption.

Assumption 2. *For the graph $\mathcal{G}(V,E)$ with n nodes, the number of edges from each node is $\mathcal{O}(1)$. Thus the number of positive links $|E|\in\mathcal{O}(n)$. And the number of negative links $(|\mathcal{U}|-|E|)\in\mathcal{O}(n^2)$. Thus the number of negative links is much greater than the number of positive links for a large n. If the predictions $\widehat{y}$ are close to the true values, we would expect the number of negative predictions ($\widehat{o}=0$) to also be much larger than the number of positive predictions ($\widehat{o}=1$). So we assume that the contribution of positive predictions to the bias is negligible.*

Let $\mathcal{U}'=\mathcal{U}\setminus E$. Under Assumption 2, the biases are

$$\text{B}(\widehat{R}_{\text{naive}}) \approx \frac{\Delta}{|\mathcal{U}'|}\left|\sum_{(i,j)\in\mathcal{U}'} y_{ij}(1-\pi_{ij})\right|,$$

$$\text{B}(\widehat{R}_{w}) \approx \text{B}(\widehat{R}_{\text{PU}}) \approx \frac{\Delta}{|\mathcal{U}'|}\left|\sum_{(i,j)\in\mathcal{U}'} y_{ij}\left(1-\frac{\pi_{ij}}{\widehat{\pi}_{ij}}\right)\right|,$$

$$\text{B}(\widehat{R}_{\text{AP}}) \approx \frac{\Delta}{|\mathcal{U}'|}\left|\sum_{(i,j)\in\mathcal{U}'}[(1-\pi_{ij})y_{ij}-(1-\pi_{ij}y_{ij})\tau_{ij}]\right|.$$

Theorem 3.2 (Comparison of Biases). *Under these approximations, a sufficient condition for $B(\widehat{R}_w) = B(\widehat{R}_{PU}) < B(\widehat{R}_{naive})$ is*

$$\frac{\pi_{ij}}{2-\pi_{ij}} < \widehat{\pi}_{ij} < 1,\ \forall(i,j)\in\mathcal{U},$$

and for $B(\widehat{R}_{AP}) < B(\widehat{R}_{naive})$ is

$$\frac{\pi_{ij}}{2-\pi_{ij}} < \widehat{\pi}_{ij} < 1 \text{ and } 0 < \widehat{y}_{ij} < cy_{ij},\ \forall(i,j)\in\mathcal{U}$$

$$\text{where } c = \frac{2(1-\pi_{ij})}{1-\widehat{\pi}_{ij}-\pi_{ij}y_{ij}+(2-\pi_{ij})\widehat{\pi}_{ij}y_{ij}} \geq 1.$$

Thus, if $\widehat{\pi}$ are not too-underestimated and $\widehat{y}$ are not too-overestimated, the proposed estimators will have lower bias than the naive estimator.

4. Learning Propensities and Link Probabilities

The previous section assumes *known* propensities ($\widehat{\pi}$) and link probabilities ($\widehat{y}$). We present a loss function that uses our proposed estimators from Section 3 to *learn* $\widehat{\pi}$ and $\widehat{y}$. A natural approach might be to minimize the negative log-likelihood of the observed data:

$$\widehat{\pi},\widehat{y} = \operatorname*{argmin}_{\widehat{\pi},\widehat{y}} \mathcal{L}(o|\widehat{y},\widehat{\pi}),$$

where $\mathcal{L}(o|\widehat{y},\widehat{\pi}) = \sum_{(i,j)\in\mathcal{U}} -o_{ij}\log(\widehat{y}_{ij}\widehat{\pi}_{ij}) - (1-o_{ij})\log(1-\widehat{y}_{ij}\widehat{\pi}_{ij})$. However, this might not ensure that the true risk remains small. We derive a generalization bound that motivates a different loss function (see Appendix B for the proof).

Definition 2 (Rademacher Complexity). *Let $\mathcal{F}$ be a class of functions $(\widehat{\pi}, \widehat{y})$. Each estimator $\widehat{R} \in \left\{\widehat{R}_w, \widehat{R}_{PU}, \widehat{R}_{AP}\right\}$ can be written as $\frac{1}{|\mathcal{U}|}\sum_{(i,j)\in\mathcal{U}} r(o_{ij}, \widehat{\pi}_{ij}, \widehat{y}_{ij})$ for an appropriate function r (e.g. by Eq. 2, for $\widehat{R}_w$, we have $r(o_{ij}, \widehat{\pi}_{ij}, \widehat{y}_{ij}) = w_{ij}\delta(o_{ij}, \widehat{y}_{ij})$) . For $\widehat{R} \in \left\{\widehat{R}_w, \widehat{R}_{PU}, \widehat{R}_{AP}\right\}$, we define a quantity analogous to the Empirical Rademacher Complexity (Bartlett & Mendelson, 2002) as*

$$\widehat{\mathcal{G}}_o(\mathcal{F}, \widehat{R}) = \mathbf{E}_\sigma \left[\sup_{(\widehat{\pi},\widehat{y})\in\mathcal{F}} \frac{1}{|\mathcal{U}|} \sum_{(i,j)\in\mathcal{U}} \sigma_{ij} r(o_{ij}, \widehat{\pi}_{ij}, \widehat{y}_{ij}) \right],$$

where σ_{ij} are independent Rademacher random variables. And the Rademacher Complexity is $\mathcal{G}(\mathcal{F}, \widehat{R}_w) = \mathbf{E}_o[\widehat{\mathcal{G}}_o(\mathcal{F}, \widehat{R}_w)]$.

$\widehat{\mathcal{G}}_o(\mathcal{F}, \widehat{R}_w)$ can be estimated from the data by taking a random sample of the variables σ_{ij} and optimizing the above objective. Next, we present a generalization bound based on $\widehat{\mathcal{G}}_o(\mathcal{F}, \widehat{R}_w)$.

Theorem 4.1 (Generalization Bound). *Let $\mathcal{F}$ be a class of functions $(\widehat{\pi}, \widehat{y})$. Let $\delta(o_{ij}, \widehat{y}_{ij}) \leq \eta \; \forall (i,j) \in \mathcal{U}$ and $\widehat{\pi}_{ij} \geq \epsilon > 0 \; \forall (i,j) \in \mathcal{U}$. Then, for $\widehat{R} \in \left\{\widehat{R}_w, \widehat{R}_{PU}, \widehat{R}_{AP}\right\}$, with probability at least $1-\delta$, we have*

$$\begin{aligned} R(\widehat{y}) &\leq \widehat{R}(\widehat{y}, \widehat{\pi}) + B(\widehat{R}) + 2\mathcal{G}(\mathcal{F}, \widehat{R}) + M \\ &\leq \widehat{R}(\widehat{y}, \widehat{\pi}) + B(\widehat{R}_w) + 2\widehat{\mathcal{G}}(\mathcal{F}, \widehat{R}_w) + 3M, \end{aligned}$$

where $M = \sqrt{\frac{4\eta^2}{\epsilon^2|\mathcal{U}|}\log(\frac{2}{\delta})}$ and $B(\widehat{R})$ is the bias of $\widehat{R}$ derived in Section 3.

Loss Function. The bound shows that $\widehat{R} \in \{\widehat{R}_w, \widehat{R}_{\text{PU}}, \widehat{R}_{\text{AP}}\}$ is close to the true risk R. This suggests that we should choose $\widehat{\pi}, \widehat{y}$ that lead to small values of $\widehat{R}$ as this will also minimize the true risk with high probability. This motivates us to learn $\widehat{\pi}, \widehat{y}$ by minimizing the following objective:

$$\widehat{\pi}, \widehat{y} = \operatorname*{argmin}_{\widehat{\pi},\widehat{y}} \mathcal{L}(o|\widehat{y}, \widehat{\pi}), \text{ subject to } \widehat{R}(\widehat{\pi}, \widehat{y}) \leq c,$$

where $\widehat{R} \in \{\widehat{R}_w, \widehat{R}_{\text{PU}}, \widehat{R}_{\text{AP}}\}$ and $c > 0$ is some constant. In practice, we minimize the following relaxed version of this objective:

$$l(\widehat{\pi}, \widehat{y}) = \lambda_L \mathcal{L}(o|\widehat{\pi}, \widehat{y}) + \lambda_R \widehat{R}(\widehat{\pi}, \widehat{y}), \tag{4}$$

where λ_R and λ_L are hyperparameters. One might try to minimize the loss function using only $\widehat{R}$ by setting $\lambda_L = 0$. This will not work because trivial solutions exist for all three risk functions: if $\forall (i,j) \in \mathcal{U},\; \widehat{y}_{ij} = 1$, then $\widehat{R}_w(\widehat{y}, \widehat{\pi}) = 0$; if $\forall (i,j) \in \mathcal{U},\; \widehat{\pi}_{ij} = 1, \widehat{y}_{ij} > 0.5$, then $\widehat{R}_{\text{PU}} = \widehat{R}_{\text{AP}} = 0$. Hence, we need to use $\lambda_L > 0$ during training to prevent the model from collapsing to these solutions. It is possible use parametric models like neural networks for $\widehat{y}$ and $\widehat{\pi}$ to incorporate information associated with the nodes (like user data or paper data). The parameters can be learned by using gradient-based methods by minimizing the loss in Eq. 4.

5. Feedback Loops

In this section, we analyze what happens when we train an RS repeatedly on data generated by users interacting with that system's recommendations. We show that for an RS that does not account for exposure bias, the fraction of high-propensity nodes that are recommended continually increases over time. In other words, the system will progressively recommend fewer low-propensity nodes, even if they are relevant, as time goes on. Next, we show that correcting for exposure bias ensures that relevant low-propensity nodes keep being recommended. In this section, we assume that the attributes of the nodes take values in a discrete set.

Assumption 3. *Each node belongs to one of C categories from the set $\mathcal{C} = \{c_1, \ldots, c_C\}$. Each category contains n nodes. $V = \{v_1, \ldots, v_N\}$ is the set of nodes and $N = nC$. The function $\gamma : V \to \mathcal{C}$ maps a node to its category. The link probability y_{ij} and propensity π_{ij} depend only on the categories of the nodes, i.e., $y_{ij} = y_{lm}$ and $\pi_{ij} = \pi_{lm}$ if $\gamma(v_i) = \gamma(v_l)$ and $\gamma(v_j) = \gamma(v_m)$. Therefore, for any pair of nodes (v_i, v_j), the product $\pi_{ij}y_{ij}$ depends only on the categories v_i and v_j belong to. Let $q_{uv} = \pi_{ij}y_{ij}$ for some v_i, v_j s.t. $\gamma(v_i) = c_u$ and $\gamma(v_j) = c_v$.*

Iterative Training Process. We now describe the iterative training process for an RS that does not account for exposure bias. We will restrict our attention to analyzing the recommendations made for the n nodes in some category $c_u \in \mathcal{C}$. We assume that we make one recommendation for each node (this simplifies exposition but is not necessary). At time step t, the fraction of nodes recommended from each category is represented by the $(C-1)$-simplex $\kappa^{(t)}$. So out of the n nodes from c_u, $n\kappa_v^{(t)}$ of them are recommended a node from the category c_v, where $\kappa_v^{(t)}$ is the v^{th} element of $\kappa^{(t)}$. Links are generated from the recommended nodes according to ground-truth propensities and link probabilities. Thus a node from c_u creates a link to a recommended node from c_v with probability q_{uv}. Since we are examining recommendations for category c_u, we will drop the subscript u going forward, i.e., $q_v = q_{uv}$. This gives us training data for the next iteration. We assume that a node only creates a link to nodes from the recommended nodes. In other words, links are not created to nodes that are not recommended.

The number of nodes linked to from category c_v at time t is $n_v^{(t)}$. Then $n_v^{(t)} \sim \text{Binomial}(n\kappa_v^{(t)}, q_v)$. During training, the link probability is estimated as $\widehat{q}_v^{(t)} = \frac{n_v^{(t)}}{n}$. We assume that, at time step $t+1$, nodes from category c_v are recommended with probability proportional to $\widehat{q}_v^{(t)}$. This is akin to recommending with some exploration (Kawale et al., 2015). Let the $(C-1)$-simplex denoting normalized estimates be $\widehat{e}^{(t+1)} = \left[\frac{\widehat{q}_1^{(t)}}{S}, \frac{\widehat{q}_2^{(t)}}{S}, \ldots, \frac{\widehat{q}_C^{(t)}}{S}\right]$, where $S = \sum_{j=1}^{C} \widehat{q}_j^{(t)}$. Thus the recommendations for the next step $\kappa^{(t+1)}$ have the distribution $\kappa^{(t+1)} \sim \frac{1}{n}\textit{Multinomial}(n, \widehat{e}^{(t+1)})$. This process is repeated at each time step. The initial training data is generated by the user generating links according to the ground-truth propensities and link probabilities.

Example 2. *We illustrate the iterative training process with a minimal example. Let $\mathcal{C} = \{c_1, c_2\}$. We examine the recommendations made to nodes in c_1. Let $n = 100, q_{1,1} = 0.8$ and $q_{1,2} = 0.4$. At time t, let $\kappa^{(t)} = [0.6, 0.4]$. Informally, 60 of the recommended nodes are from c_1 and the remaining 40 from c_2. The nodes create links to the recommended nodes with probabilities $q_{1,1}$ and $q_{1,2}$. Therefore, the number of nodes linked that belong to c_1 at time t is $n_1^{(t)} \sim$ Binomial$(60, q_1 = 0.8)$ and similarly, $n_2^{(t)} \sim$ Binomial$(40, q_2 = 0.4)$. Informally, the realized values are $n_1^{(t)} = 48$ and $n_2^{(t)} = 16$. The estimated link probabilities are $\widehat{q}_1^{(t)} = 0.48, \widehat{q}_2^{(t)} = \frac{16}{100} = 0.16$ and $\widehat{e}^{(t+1)} = \left[\frac{0.48}{0.64}, \frac{0.16}{0.64}\right] = [0.75, 0.25]$. Then, at time $t+1$, we recommend nodes according to $\widehat{e}^{(t+1)}$, i.e., $\kappa^{(t+1)} \sim \frac{1}{100}$Multinomial$(100, \widehat{e}^{(t+1)})$. The realized value of $\kappa_1^{(t+1)}$ is likely to be greater than $\kappa_1^{(t)}$. Thus more items from c_1 are likely to be recommended at time $t+1$ as compared to time t. This provides some intuition for the existence of a feedback loop: nodes that are linked less are in turn recommended with a lower probability in the next time step.*

We formally show the existence of feedback loops (see Appendix C for the proofs). We prove a finite-sample result which shows that, with high probability, the relative probability of recommending nodes from categories with higher values of q_j keeps increasing over time.

Theorem 5.1. *Suppose that $q_v > q_w$ if $v > w$. Let $\kappa_{vw}^{(t)} = \frac{\kappa_v^{(t)}}{\kappa_v^{(t)} + \kappa_w^{(t)}}$. Let $A_{vw}^{(t)}$ represent the event that relative fraction of recommendations from c_v to that from c_w increases at time t, i.e., $\kappa_{vw}^{(t+1)} > \kappa_{vw}^{(t)}$. Let $A^{(t)}$ be the event that all relative fractions get skewed towards c_v from c_w if $q_v > q_w$, i.e., $A^{(t)} = \bigcap_{(v,w)\in\mathcal{S}} A_{vw}^{(t)}$, where $\mathcal{S} = \{(v,w) : v \in [C], w \in [C], v > w\}$. Then, for constants $\epsilon, \eta > 0$ that only depend on $\kappa^{(t)}$ and q, we have*

$$\begin{aligned}\mathbf{P}(A^{(t)}|\kappa^{(t)}) &\geq 1 - 2C\exp\left(-2n\left[\epsilon^2 + \frac{\eta^2}{C^2}\right]\right)\\ &\geq 1 - 2C\exp\left(-\mathcal{O}\left(\frac{n}{C^2}\right)\right).\end{aligned}$$

Corollary 5.1. $\lim_{n\to\infty} \mathbf{P}(A^{(t)}|\kappa^{(t)}) = 1$ *if* $C^3 \in o(n)$.

Thus, at each time step, with high probability, nodes with low propensity are less likely to be recommended in the next time step. Therefore, if an RS does not correct for exposure bias, over time, even relevant nodes with low propensity are unlikely to be recommended. Next, we derive and analyze the rate at which the exposure bias exacerbates.

Theorem 5.2. *Let $q_v > q_w$. As $n \to \infty$, $\kappa_{vw}^{(t)} \xrightarrow{p} 1 - \frac{1}{1+c^t}$, where $c = \frac{q_v}{q_w}$.*

Theorem 5.2 shows that the rate at which the bias exacerbates is dependent on the ratio $\frac{q_v}{q_w}$. Therefore, the lower the propensity, the faster the probability of that node being recommended reduces.

Corollary 5.2. *Let $y_{c_u c_v} = y_{ij}$ for some (i,j) s.t. $u = \gamma(i)$ and $v = \gamma(j)$ (γ is defined after Assumption 3). We now assume that we have a consistent estimator $\widehat{q}_v^{(t)} \xrightarrow{p} \kappa_v^{(t)} y_{g_u g_v}$, where $\kappa_v^{(t)}$ is the v^{th} element of the simplex $\kappa^{(t)}$. Thus $\widehat{q}_v^{(t)}$ is an estimator that negates the effect of exposure bias. As $n \to \infty$, $\kappa_{vw}^{(t)} \xrightarrow{p} 1 - \frac{1}{1+c^t}$, where $c = \frac{y_{g_u g_v}}{y_{g_u g_w}}$.*

This shows that accounting for exposure bias can alleviate the feedback loop. Despite having low propensity, relevant papers will continue to be recommended.

6. Experiments

We validate our link recommendation methods on the task of citation recommendation. Given an input paper's data (like title, abstract, etc.), the goal is to recommend papers that it should cite. We use the Microsoft Academic Graph (MAG) dataset (Sinha et al., 2015). MAG is a graph containing scientific papers and the citation relationships between them. It also contains the titles, abstracts, and FOS of the papers. In our experiments, we use subgraphs from the MAG by performing a breadth-first search from some root node. For each paper, we concatenate the title and abstract and generate a 768-dimensional embedding for the text using the *bert-as-service* library (Xiao, 2018). We use a SciBERT model (Beltagy et al., 2019), which is a BERT model trained on scientific text, with this library. For each paper p_i, we generate the embedding $h_i \in \mathbb{R}^{768}$. The FOS in MAG are organised as a tree, where a child is a sub-field of its parent. We only use the root-level FOS for each paper and there are 19 such FOS. We use Amazon Sagemaker (Liberty et al., 2020) to run our experiments.

Table 1. Evaluation metrics on the test set of the semi-synthetic data computed against known ground truth citation links.

MODEL	PREC.	REC.	AUC	MAP
NO PROP.	67.24	54.81	84.45	41.87
MLE	81.04	60.19	93.12	56.77
$\widehat{R}_w$	**83.28**	63.73	**96.42**	56.96
$\widehat{R}_{\text{PU}}$	82.16	63.07	94.28	58.01
$\widehat{R}_{\text{AP}}$	83.01	**65.54**	95.38	**59.90**

For simplicity, we assume that the propensities π_{ij} depend only on the FOS of papers p_i and p_j. Thus the propensity model is parameterized by $\widehat{\theta}_\pi \in [0,1]^{19\times 19}$. However, our methods can easily extend to more complicated parametric propensity estimators like neural networks. To model the link probability $\widehat{y}_{ij}$, we use the following model:

$$\widehat{y}_{ij} = \sigma(\widehat{w}^\top (h_i \odot h_j) + \widehat{b}), \tag{5}$$

where $\widehat{w} \in \mathbb{R}^{768}$ and $\widehat{b} \in \mathbb{R}$ are trainable parameters, $\odot$ is an element-wise product, and σ is the sigmoid function. We use stochastic gradient descent to learn $\widehat{\theta}_\pi, \widehat{w}$ and $\widehat{b}$ using the loss function described in Eq. 4 with δ as the log-loss, i.e., $\delta(u, \widehat{u}) = -u \log(\widehat{u}) - (1-u)\log(1-\widehat{u})$. For training, we use the Adam optimizer (Kingma & Ba, 2014) with a learning rate of 10^{-4} and a batch size of 32.

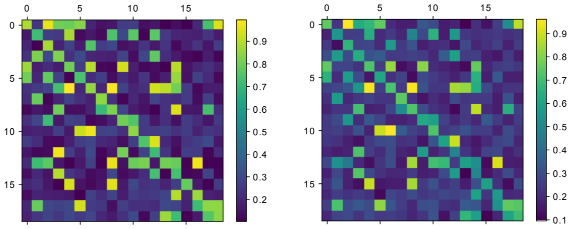
6.1. Semi-Synthetic Dataset

Since we do not have ground truth exposure values in the MAG dataset, we cannot know whether a paper was not cited due to a lack of exposure or due to irrelevancy. As a result, we construct a semi-synthetic dataset with simulated propensity scores and link probabilities. We use a subset of 41,600 papers. We generate train-test-validation splits by taking a topological ordering of the nodes and use the subgraph created from the first 70% for training, next 10% for validation, and the remaining 20% for testing. We use the real text and FOS for each paper. The simulated propensity matrix π is a 19×19 matrix with its diagonal and off-diagonal entries initialized from $U(0.7, 1)$ and $U(0.1, 0.3)$, respectively, where $U(.)$ is the uniform distribution. The link probability is simulated using $y_{ij} = \sigma(w^\top (h_i \odot h_j) + b)$, where σ is the sigmoid function, $\odot$ is element-wise product, and w, b are fixed *known* vectors.

We show the evaluation metrics for five models on the test set computed against the simulated true citations (not the observed citations) (Table 1). *No Prop* is the model trained naively on the observed data using only the output model in Eq. 5. *MLE* is the model trained using the loss function in Eq. 4 with $\lambda_R = 0$. The remaining three are models trained using $\widehat{R}_w$, $\widehat{R}_{\text{PU}}$, and $\widehat{R}_{\text{AP}}$ with $\lambda_R = 10$ and $\lambda_L = 1$. We see that all other estimators significantly outperform *No Prop*. Additionally, our proposed estimators lead to improved performance over the MLE. We emphasize that these metrics are computed against true citations and thus are a measure of true risk which is the appropriate metric for evaluating an RS's performance. This shows the utility of accounting for exposure bias and learning using our proposed loss function.

Table 2. RMSE of the estimated risk with respect to the true risk computed using our proposed estimators. The first column shows the risk used in the loss function in Eq. 4 to learn $\widehat{\pi}$ and $\widehat{y}$.

TRAINED USING	ESTIMATOR USED $\widehat{R}_{\text{NAIVE}}$	$\widehat{R}_w$	$\widehat{R}_{\text{PU}}$	$\widehat{R}_{\text{AP}}$
NO PROP.	1.50	-	-	-
MLE	0.67	**0.23**	0.24	0.32
$\widehat{R}_w$	0.43	**0.04**	0.10	0.11
$\widehat{R}_{\text{PU}}$	0.38	0.05	0.11	**0.04**
$\widehat{R}_{\text{AP}}$	0.41	0.06	0.08	**0.03**



(a) Ground Truth

(b) Estimated

Figure 1. The estimated propensities propensities are close to the true simulated values when learned using $\widehat{R}_w$.

In this work, we tackle two separate (but related) challenges. The first challenge is *learning* link probabilities in such a way that they are not underestimated due to exposure bias. The second challenge is *evaluating* a RS given learned link probabilities and propensity scores, i.e., computing a good estimate of the true risk. We demonstrate the efficacy of our methods for the second challenge and show that our proposed weighting schemes lead to good estimates of the true risk (Table 2). We show the RMSE of the risk estimated using the proposed estimators with respect to the true risk. The first column denotes risk function used to train the model (as described in Section 4). The rest of the columns denote the estimators used to estimate true risk using the learned propensities and link probabilities from the trained model in the first column. We trained each model 10 times to compute the RMSE. The RMSE estimated using $\widehat{R}_{\text{naive}}$ is always greater than that of the other estimators, which shows that leveraging the learned propensities leads to substantially better estimates of the true risk (and thus more accurately evaluates the RS). The RMSE when trained using the *MLE* is higher than when trained using our proposed estimators, showing the benefit of our proposed estimators over the *MLE*. This also qualitatively validates the generalization

Table 3. Evaluation metrics for various models computed on the test sets of the two real-world citation datasets.

MODEL	PREC.	REC.	F1	AUC	MAP
DATASET 1					
NO PROP.	29.45	78.30	42.81	84.44	24.10
MLE	30.24	77.84	43.56	84.41	24.60
$\widehat{R}_w$	31.46	78.02	44.84	84.74	25.60
$\widehat{R}_{\text{PU}}$	30.98	**78.94**	44.49	**85.24**	25.11
$\widehat{R}_{\text{AP}}$	**36.07**	76.08	**48.94**	84.67	**28.58**
DATASET 2					
NO PROP.	44.86	70.85	54.94	83.22	33.19
MLE	44.43	74.66	55.71	84.97	34.39
$\widehat{R}_w$	**48.70**	71.62	**57.98**	83.90	**36.25**
$\widehat{R}_{\text{PU}}$	42.17	**76.15**	54.28	**85.43**	33.26
$\widehat{R}_{\text{AP}}$	47.22	71.84	56.98	83.89	35.27

Table 4. Link prediction metrics for various models when evaluated on the test set of the real datasets.

MODEL	RECALL @100	MEAN RANK	ENTROPY@100 TRUE POSITIVES
DATASET 1			
NO PROP.	24.39	**2247.27**	1.65
MLE	24.70	2891.40	1.73
$\widehat{R}_w$	25.03	2836.73	**1.74**
$\widehat{R}_{\text{PU}}$	24.61	2875.13	1.73
$\widehat{R}_{\text{AP}}$	**26.66**	2425.51	1.71
DATASET 2			
NO PROP.	**6.32**	**10170.26**	1.06
MLE	6.07	10731.88	1.08
$\widehat{R}_w$	6.30	10873.19	**1.12**
$\widehat{R}_{\text{PU}}$	5.92	10717.06	1.08
$\widehat{R}_{\text{AP}}$	5.99	10801.11	1.10

bound proved in Section 4 by showing that minimizing $\widehat{R} \in \{\widehat{R}_w, \widehat{R}_{\text{PU}}, \widehat{R}_{\text{AP}}\}$ also leads to small values of the true risk.

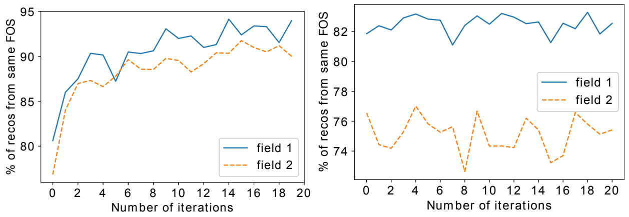
The heatmap of simulated propensities and estimated propensities when using $\widehat{R}_w$ shows that the estimated propensities are close to the true propensities (Figure 1). The mean relative error between the true and estimated propensities is 19.47%, demonstrating that the training procedure recovers the propensities. Together, these results show that our methods successfully mitigate exposure bias in this dataset.

6.2. Real-World Datasets

We now evaluate our proposed method on a real-world citation network. We construct two datasets by using disjoint subgraphs of the MAG. The first generated dataset has $2{,}442{,}008$ papers and $7{,}577{,}886$ edges. The second dataset has $1{,}328{,}664$ papers and $1{,}469{,}899$ edges. Thus the second graph is sparser than the first one. The FOS distribution is also different in both datasets (see details in Appendix D). We use 70-10-20% train-validation-test splits generated similarly to the semi-synthetic dataset. We do not have access to true exposure values and thus we evaluate our methods against the observed citation links.

We show the evaluation metrics for the proposed estimators (Table 3). Since we do not have access to the true citation links in the real dataset, we compute these metrics over the observed links. In other words, this is a measure of the naive risk. We see that our proposed estimators achieve comparable metrics to *No Prop*. For both datasets, the best numbers for each metric are achieved by estimators other than *No Prop*. Moreover, the models using the weighted estimators outperform the *MLE* estimator in both datasets. These results show that our proposed estimators achieve comparable performance even when evaluated on the observed citation data. Similarly, Table 4 shows link prediction metrics for various models computed against *observed* citations. *Recall@100* refers to the recall in the top 100 recommendations averaged across all papers in the test set. *Mean Rank* is the mean rank of the cited papers averaged across all the papers. *Entropy@100 of True Positives* is the entropy in the FOS of the true positives in the top 100 recommendations for each paper; we use it to measure the diversity in the FOS of the recommendations. Our proposed estimators achieve comparable *Recall@100* and *Mean Rank* to *No Prop* for both datasets. As expected, propensity based estimators have higher FOS entropy scores than *No Prop*, with $\widehat{R}_w$ achieving the highest FOS entropy in both datasets. Thus our proposed estimators *recommend more relevant papers from different FOS and still maintain comparable performance* to *No Prop*.

At first blush, the comparable performance to *No Prop* may not seem compelling. However, this is a strong result. Our goal is to correct exposure bias and minimize *true risk*, not observed (or naive) risk. Since Tables 3 and 4 are computed against observed citations, our proposed methods should not be expected to outperform *No Prop* as they are not trying to optimize metrics against the observed links. In Section 6.1, we showed that our methods correct exposure bias and achieve lower true risk. Coupled with those results, our goal in this section was to show that our methods do not lower performance even if evaluated against the standard evaluation metrics. We suspect that the negative log-likelihood term $\mathcal{L}(o|\widehat{\pi}, \widehat{y})$ in Eq. 4 is likely responsible for the comparable performance against the observed risk. This is because, as seen from the results, the MLE also performs well as compared to *No Prop*..



(a) No Propensity

(b) With Propensity ($\widehat{R}_w$)

Figure 2. The fraction of recommended papers from the same FOS over time.

6.3. Feedback Loops

We run simulations to examine what happens when a citation recommender is trained repeatedly on data collected from users interacting with its recommendations. We use the iterative training procedure described in Section 5. We construct a training set of 410 papers from the MAG, with their corresponding real FOS and text embeddings. For ease of exposition, we use an arbitrary but fixed mapping to map the 19 FOS to two FOS. The synthetic propensities and link probabilities are simulated similarly to Sec 6.1. In the first iteration, the models are trained using the observed citation network. For subsequent training iterations, the training data is generated as follows. For each paper, we recommend 20 papers. The probability of recommending a paper is proportional to its estimated citation probability. We then simulate the user's interaction with the recommendations according to the known simulated propensities and citation probabilities. This generates the training set for the subsequent iteration. We then repeat this process.

We show how the fraction of recommended papers from the same FOS changes over multiple training iterations for models trained without propensity, i.e., *No Prop* and the model trained using $\widehat{R}_w$ (Figure 2). We plot this time series for both FOS in our dataset. For *No Prop*, the fraction of papers recommended from the same FOS increases over time for both FOS (Figure 2a). This demonstrates the existence of a feedback loop that worsens exposure bias and reduces the number of papers recommended from a different FOS over time. On the other hand, when we train our models using $\widehat{R}_w$, the feedback loop no longer exists and the fraction of papers recommended from a different FOS remains stable over time (Figure 2b). This shows that our proposed estimator continues to recommend relevant papers from a different FOS and corrects the feedback loop.

7. Conclusion

Proposing three estimators to correct for exposure bias, we derive sufficient conditions for when they exhibit lower bias than the naive estimator and incorporate them into a learning procedure. Theoretically, we prove that feedback loops can worsen exposure bias. Empirically, we show that proposed estimators improve performance against the true link probabilities, leading to better estimates of true risk, and combating feedback loops. Our methods can be extended to RSs that use different propensity or link probability models. Using domain knowledge (e.g., through graphical models) to improve propensity learning and empirically evaluating our methods in other link recommendation tasks are promising future directions. Exposure bias in link recommendation also raises fairness concerns. For example, in citation recommendation, certain authors or institutions might get unfair exposure which can be worsened by the RS. Investigating exposure bias correction methods for providing fairer recommendations would also be interesting future work.

References

Bai, X., Wang, M., Lee, I., Yang, Z., Kong, X., and Xia, F. Scientific paper recommendation: A survey. *IEEE Access*, 2019.

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 2002.

Beel, J., Gipp, B., Langer, S., and Breitinger, C. Paper recommender systems: a literature survey. *International Journal on Digital Libraries*, 2016.

Bekker, J., Robberechts, P., and Davis, J. Beyond the selected completely at random assumption for learning from positive and unlabeled data. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2019.

Beltagy, I., Lo, K., and Cohan, A. Scibert: Pretrained language model for scientific text. In *Empirical Methods in Natural Language Processing*, 2019.

Brandão, M. A., Moro, M. M., Lopes, G. R., and Oliveira, J. P. Using link semantics to recommend collaborations in academic social networks. In *Proceedings of the 22nd International Conference on World Wide Web*, 2013.

Chaney, A. J., Stewart, B. M., and Engelhardt, B. E. How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. In *ACM Conference on Recommender Systems*, 2018.

Chang, J. and Blei, D. Relational topic models for document networks. In *Artificial Intelligence and Statistics*, 2009.

Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., and He, X. Bias and debias in recommender system: A survey and future directions. *arXiv preprint arXiv:2010.03240*, 2020.

Haruna, K., Akmar Ismail, M., Suhendroyono, S., Damiasih, D., Pierewan, A. C., Chiroma, H., and Herawan, T. Context-aware recommender system: A review of recent developmental process and future research direction. *Applied Sciences*, 2017.

Jeunen, O. Revisiting offline evaluation for implicit-feedback recommender systems. In *ACM Conference on Recommender Systems*, 2019.

Jiang, R., Chiappa, S., Lattimore, T., György, A., and Kohli, P. Degenerate feedback loops in recommender systems. In *AAAI/ACM Conference on AI, Ethics, and Society*, 2019.

Joachims, T., Swaminathan, A., and Schnabel, T. Unbiased learning-to-rank with biased feedback. In *ACM International Conference on Web Search and Data Mining*, 2017.

Kawale, J., Bui, H. H., Kveton, B., Tran-Thanh, L., and Chawla, S. Efficient thompson sampling for online matrix-factorization recommendation. In *Advances in Neural Information Processing Systems*, 2015.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Li, Z., Fang, X., and Sheng, O. R. L. A survey of link recommendation for social networks: methods, theoretical foundations, and future research directions. *ACM Transactions on Management Information Systems (TMIS)*, 2017.

Liang, D., Charlin, L., and Blei, D. M. Causal inference for recommendation. In *Causation: Foundation to Application, Workshop at UAI*. AUAI, 2016a.

Liang, D., Charlin, L., McInerney, J., and Blei, D. M. Modeling user exposure in recommendation. In *International Conference on World Wide Web*, 2016b.

Liberman, S. and Wolf, K. B. Independent simultaneous discoveries visualized through network analysis: the case of linear canonical transforms. *Scientometrics*, 2015.

Liberty, E., Karnin, Z., Xiang, B., Rouesnel, L., Coskun, B., Nallapati, R., Delgado, J., Sadoughi, A., Astashonok, Y., Das, P., et al. Elastic machine learning algorithms in amazon sagemaker. In *ACM SIGMOD International Conference on Management of Data*, 2020.

Ma, S., Zhang, C., and Liu, X. A review of citation recommendation: from textual content to enriched context. *Scientometrics*, 2020.

Ma, W. and Chen, G. H. Missing not at random in matrix completion: The effectiveness of estimating missingness probabilities under a low nuclear norm assumption. In *Advances in Neural Information Processing Systems*, 2019.

Mansoury, M., Abdollahpouri, H., Pechenizkiy, M., Mobasher, B., and Burke, R. Feedback loop and bias amplification in recommender systems. *arXiv preprint arXiv:2007.13019*, 2020.

Masrour, F., Wilson, T., Yan, H., Tan, P.-N., and Esfahanian, A. Bursting the filter bubble: Fairness-aware network link prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

Schnabel, T., Swaminathan, A., Singh, A., Chandak, N., and Joachims, T. Recommendations as treatments: debiasing learning and evaluation. In *International Conference on Machine Learning*, 2016.

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Sinha, A., Shen, Z., Song, Y., Ma, H., Eide, D., Hsu, B.-J., and Wang, K. An overview of microsoft academic service (mas) and applications. In *International Conference on World Wide Web*, 2015.

Sinha, A., Gleich, D. F., and Ramani, K. Deconvolving feedback loops in recommender systems. In *Advances in Neural Information Processing Systems*, 2016.

Sun, W., Khenissi, S., Nasraoui, O., and Shafto, P. Debiasing the human-recommender system feedback loop in collaborative filtering. In *Companion Proceedings of The 2019 World Wide Web Conference*, 2019.

Swaminathan, A. and Joachims, T. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning*, 2015.

Vincenot, C. E. How new concepts become universal scientific approaches: insights from citation network analysis of agent-based complex systems science. *The Royal Society B: Biological Sciences*, 2018.

Wang, A. and Russakovsky, O. Directional bias amplification. *arXiv preprint arXiv:2102.12594*, 2021.

Wang, H. and Li, W.-J. Online egocentric models for citation networks. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

Wang, H. and Yeung, D.-Y. Towards bayesian deep learning: A framework and some existing methods. *TDKE*, 28(12): 3395–3408, 2016.

Wang, H. and Yeung, D.-Y. A survey on bayesian deep learning. *ACM Computing Surveys (CSUR)*, 53(5):1–37, 2020.

Wang, H., Wang, N., and Yeung, D. Collaborative deep learning for recommender systems. In *KDD*, pp. 1235–1244, 2015.

Wang, H., Shi, X., and Yeung, D.-Y. Relational deep learning: A deep latent variable model for link prediction. In *Association for the Advancement of Artificial Intelligence*, 2017.

Wang, J., Zhu, L., Dai, T., and Wang, Y. Deep memory network with bi-lstm for personalized context-aware citation recommendation. *Neurocomputing*, 2020a.

Wang, L., Bai, Y., Sun, W., and Joachims, T. Fairness of exposure in stochastic bandits. *arXiv preprint arXiv:2103.02735*, 2021.

Wang, Y., Liang, D., Charlin, L., and Blei, D. M. Causal inference for recommender systems. In *ACM Conference on Recommender Systems*, 2020b.

Xiao, H. bert-as-service. `https://github.com/hanxiao/bert-as-service`, 2018.

Yang, L., Cui, Y., Xuan, Y., Wang, C., Belongie, S., and Estrin, D. Unbiased offline recommender evaluation for missing-not-at-random implicit feedback. In *ACM Conference on Recommender Systems*, 2018.

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., and Chang, K.-W. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. *arXiv preprint arXiv:1707.09457*, 2017.