Unbiased Implicit Recommendation and Propensity Estimation via Combinational Joint Learning

ZIWEI ZHU, Texas A&M University
YUN HE, Texas A&M University
YIN ZHANG, Texas A&M University
JAMES CAVERLEE, Texas A&M University

This paper focuses on how to generate unbiased recommendations based on biased implicit user-item interactions. We propose a combinational joint learning framework to simultaneously learn unbiased user-item relevance and unbiased propensity. More specifically, we first present a new unbiased objective function for estimating propensity. We then show how a naïve joint learning approach faces an estimation-training overlap problem. Hence, we propose to jointly train multiple sub-models from different parts of the training dataset to avoid this problem. Finally, we show how to incorporate residual components trained by the complete training data to complement the relevance and propensity sub-models. Extensive experiments on two public datasets demonstrate the effectiveness of the proposed model with an improvement of 4% on average over the best alternatives.

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

ACM Reference Format:
Ziwei Zhu, Yun He, Yin Zhang, and James Caverlee. 2020. Unbiased Implicit Recommendation and Propensity Estimation via Combinational Joint Learning. In *Fourteenth ACM Conference on Recommender Systems (RecSys '20), September 22–26, 2020, Virtual Event, Brazil*. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3383313.3412210

1 INTRODUCTION

Widespread implicit user-item interactions such as user purchases, views, and clicks, have been widely used in recommender systems (RecSys) with far-reaching impact [5, 6, 12]. However, recent studies [2, 13, 15, 22] show that these implicit interactions are not necessarily aligned with user preferences. Since the observed interactions are determined by both *user-item relevance* and *item exposure*, learning a model directly from implicit interactions results in a biased RecSys.

To address this issue, recent works have proposed unbiased recommendation models by applying principles from Inverse Propensity Scoring (IPS) [13, 15]. These unbiased algorithms can theoretically guarantee to produce unbiased user-item relevance prediction conditioned on having an unbiased estimation of the propensity (i.e., the user-item exposure probability). However, these works model this item exposure probability by a power-law function of item popularity (e.g., using the number of interactions received by an item), which *is not an unbiased estimate of the propensity*. The item exposure probability of an item depends on the total number of users who have seen the item, which is a function of both the observed positive feedback (e.g., clicks) and the *unobserved negative feedback*. That is, some users may see an item but not interact with it. This unobserved negative feedback is missing from existing item exposure methods based on item popularity, and so bias may still be introduced into seemingly unbiased methods.

Therefore, we propose a combinational joint learning framework that is designed to simultaneously learn unbiased user-item relevance and unbiased propensity. More specifically, we first introduce an unbiased propensity estimation method that aims to learn the unbiased user-item exposure probability directly from observed user-item interaction records, rather than assuming a power law distribution. Such an approach has the benefit of learning propensity directly from data, sidestepping the disadvantages of heuristic methods used in previous works. Because learning unbiased relevance and learning unbiased propensity are conditioned on each other, a straightforward way for unbiased recommendation is to learn both of them via a joint learning model. However, we show how a naïve joint learning method that iteratively train an unbiased relevance model and an unbiased propensity model can still lead to a special *estimation-training overlap* problem, wherein the learning of the relevance and propensity models shares the same training data, leading to biased results. Hence, we propose a new combinational joint learning framework that jointly learns multiple unbiased relevance and propensity sub-models from different parts of the training dataset to avoid this estimation-training overlap problem. We further show how to incorporate residual components trained by the complete training data to complement these relevance and propensity sub-models, leading to unbiased prediction of user-item relevance and propensity. By experiments on two real-world datasets, we show how the proposed model improves existing unbiased recommendation methods with an improvement of 4% on average over the best alternatives.

2 RELATED WORK

Prior works have studied feedback bias in explicit ratings in RecSys [4, 9, 14, 16, 17, 19–21], especially since people can be selective for which items to provide ratings [10, 16–19]. As a result, many approaches are inherently biased, with more accurate predictions for high ratings than for low ratings. To address this, previous works propose to alleviate bias in terms of both model learning [4, 9, 14, 16, 17, 20, 21, 23] and evaluation metrics [16, 17, 19].

With increasing impact made by implicit RecSys, investigating the bias in implicit feedback is in high demand. Yang et al. [22] studied the influence of bias in implicit feedback in term of evaluation, showing that conventional evaluation metrics are biased toward high-exposure items and proposing unbiased metrics based on IPS method. Based on the IPS method as well, Saito et al. [15] proposed the first unbiased recommendation model to learn unbiased user-item relevance from biased implicit data. Later, Saito [13] extended the point-wise model in [15] to a pair-wise version, which delivers improved performance. However, although these existing unbiased methods theoretically guarantee to generate unbiased recommendation, their reliance on naïve item popularity based estimation of propensity in the IPS method can still lead to inaccurate and biased recommendation. In this work, we address this issue by the proposed combinational joint learning method to learn both unbiased relevance and unbiased propensity simultaneously.

3 PROPOSED METHOD

In this section, we introduce a combinational joint learning framework that jointly learns unbiased relevance and propensity simultaneously. We begin by formalizing the implicit recommendation problem and introducing an unbiased objective function from previous work to model the unbiased user-item relevance. We then show how to estimate unbiased propensity – to overcome the hidden bias prevalent in many previous approaches to estimate propensity – and then provide the details of the combinational joint learning framework.

3.1 Preliminaries

Problem Statement. Suppose we have a user set $\mathcal{U} = \{1, 2, \ldots, N\}$, an item set $\mathcal{I} = \{1, 2, \ldots, M\}$, and a user-item interaction variable $Y_{u,i} \in \{0, 1\}$ where $u \in \mathcal{U}$ and $i \in \mathcal{I}$ recording observed interactions ($Y_{u,i} = 1$) or unknown interactions ($Y_{u,i} = 0$). We use $\mathcal{D}$ to denote the training data with all observed user-item interactions and some unknown interactions (by random negative sampling). To model the observed interaction variable, previous works [13, 15, 22] introduce two hidden variables: the relevance variable $R_{u,i} \in \{0, 1\}$ indicating whether user u likes item i ($R_{u,i} = 1$) or not ($R_{u,i} = 0$); and the exposure variable $O_{u,i} \in \{0, 1\}$ indicating whether item i is exposed to user u ($O_{u,i} = 1$) or not

($O_{u,i} = 0$). Then, the interaction variable is modeled as: $Y_{u,i} = R_{u,i} \cdot O_{u,i}$, i.e., only if user u likes item i ($R_{u,i} = 1$) and i is exposed to u ($O_{u,i} = 1$), can we observe $Y_{u,i} = 1$. Hence, the task of unbiased implicit RecSys is to infer user-item relevance $R_{u,i}$ and provide ranked lists of items to users based on the observed interaction variable $Y_{u,i}$.

Unbiased Objective Function via IPS. To model the relevance variable $R_{u,i}$, we can have the ideal objective function [15] : $\mathcal{L}_{ideal} = \sum_{(u,i)\in\mathcal{D}} R_{u,i}(log(\widehat{R}_{u,i})) + (1 - R_{u,i})(log(1 - \widehat{R}_{u,i}))$, where $\widehat{R}_{u,i}$ is the predicted relevance probability for user u to item i, which can be formulated as a matrix factorization model: $\widehat{R}_{u,i} = \sigma(\mathbf{P}_u^\top \cdot \mathbf{Q}_i)$ with $\mathbf{P}_u$ as the user latent factors, $\mathbf{Q}_i$ as the item latent factors, and $\sigma(\cdot)$ as the Sigmoid function. Note that here we adopt cross entropy, but other loss functions can be selected as well.

However, in practice, $R_{u,i}$ is unobservable. That is, we can only observe the interaction variable $Y_{u,i}$ that conflates both relevance and exposure. Conventional algorithms [3, 8, 11, 12] directly replace $R_{u,i}$ in $\mathcal{L}_{ideal}$ by $Y_{u,i}$, which is problematic because it will lead to the learned $\widehat{R}_{u,i}$ combining both relevance $R_{u,i}$ and exposure $O_{u,i}$ (because $Y_{u,i} = R_{u,i} \cdot O_{u,i}$) to generate biased recommendations. Hence, to address this problem, previous work [15] adopts Inverse Propensity Scoring (IPS), leading to the following unbiased objective function:

$$\mathcal{L}_{IPS} = \sum_{(u,i)\in\mathcal{D}} \frac{Y_{u,i}}{\theta_{u,i}}(log(\widehat{R}_{u,i})) + (1 - \frac{Y_{u,i}}{\theta_{u,i}})(log(1 - \widehat{R}_{u,i})), \tag{1}$$

where all variables $Y_{u,i}$, $R_{u,i}$, $O_{u,i}$ are assumed to be Bernoulli variables as $P(Y_{u,i} = 1) = \gamma_{u,i} \cdot \theta_{u,i}$, $\gamma_{u,i} = P(R_{u,i} = 1)$ and $\theta_{u,i} = P(O_{u,i} = 1)$. It is straightforward to show $\mathbb{E}[\mathcal{L}_{IPS}] = \mathbb{E}[\mathcal{L}_{ideal}]$ (for details, please refer to [15]). Thus, by minimizing $\mathcal{L}_{IPS}$, we can have unbiased recommendation.

3.2 Unbiased Propensity Estimation

We call the parameter $\theta_{u,i}$ (the probability of exposing item i to user u) the propensity in the IPS method. This propensity is estimated by a power-law function of item popularity (the number of interactions received by each item) in [13, 15, 22]:

$$\theta_{*,i} = (\sum_{u\in\mathcal{U}} Y_{u,i}/max_{i\in\mathcal{I}}(\sum_{u\in\mathcal{U}} Y_{u,i}))^{\eta}. \tag{2}$$

However, the power-law function of item popularity is itself not an unbiased estimation of the exposure probability: item popularity only considers the observed positive user-item interactions, but item exposure is determined by both observed positive interactions and unobserved negative feedback. That is, users may see an item but not interact with it. As a result, bias may still be introduced into seemingly unbiased methods such as in Equation 1.

Hence, we propose an unbiased propensity estimation method that aims to learn (i) the trade-off between the item popularity of positive and negative interactions; and (ii) the relative popularity for unobserved negative interactions. Such an approach has the added side benefit of learning propensity directly from data, sidestepping the challenge of tuning the exponent hyper-parameter η accurately for every new dataset.

Because all of $Y_{u,i}$, $R_{u,i}$, and $O_{u,i}$ are Bernoulli variables as introduced in Section 3.1, the unbiased objective function for modeling $O_{u,i}$ is symmetric to the unbiased objective function for $R_{u,i}$ in Equation 1. Thus, by replacing $\theta_{u,i}$ with $\gamma_{u,i}$ and replacing $\widehat{R}_{u,i}$ with $\widehat{O}_{u,i}$ in Equation 1, we have the Inverse Relevance Scoring objective function:

$$\mathcal{L}_{IRS} = \sum_{(u,i)\in\mathcal{D}} \frac{Y_{u,i}}{\gamma_{u,i}}(log(\widehat{O}_{u,i})) + (1 - \frac{Y_{u,i}}{\gamma_{u,i}})(log(1 - \widehat{O}_{u,i})), \tag{3}$$

where $\widehat{O}_{u,i}$ is the predicted exposure probability, i.e., the estimation of the propensity $\theta_{u,i}$. Concretely, we model $\widehat{O}_{u,i}$ by an item-based model: $\widehat{O}_{u,i} = (w \cdot a + (1 - w) \cdot K_i)^e$, where $w = f_w(\mathbf{Q}_i)$, $a = f_a(\mathbf{Q}_i)$, $e = f_e(\mathbf{Q}_i)$; $f_w(\cdot)$, $f_a(\cdot)$, $f_e(\cdot)$

Algorithm 1: Training algorithm.

repeat
  for $\mathcal{D}_c$ *in* $\{\mathcal{D}_1, \ldots, \mathcal{D}_C\}$ **do**
    for (u, i) *in* $\mathcal{D}_c$ **do**
      Calculate $\gamma_{u,i}$ and $\theta_{u,i}$ by Ψ_c and Φ_c;
      Update $\{\Psi_1, \ldots, \Psi_C\} \setminus \Psi_c$ by Equation 1, and update $\{\Phi_1, \ldots, \Phi_C\} \setminus \Phi_c$ by Equation 3;
      with $\{\Psi_1, \ldots, \Psi_C\}$ and $\{\Phi_1, \ldots, \Phi_C\}$ fixed:
        Update $\{\overline{\Psi}_1, \ldots, \overline{\Psi}_C\}$ by Equation 1 with $\widehat{R}_{u,i}$ calculated by $\{\Psi_1 + \overline{\Psi}_1, \ldots \Psi_C + \overline{\Psi}_C\}$;
        Update $\{\overline{\Phi}_1, \ldots, \overline{\Phi}_C\}$ by Equation 3 with $\widehat{O}_{u,i}$ calculated by $\{\Phi_1 + \overline{\Phi}_1, \ldots, \Phi_C + \overline{\Phi}_C\}$;

until ***converge***;

are three one layer perceptrons activated by a Sigmoid function with the item latent vector as input and a probability scalar as output; $K_i = \sum_{u \in \mathcal{U}} Y_{u,i} / max_{i \in \mathcal{I}} (\sum_{u \in \mathcal{U}} Y_{u,i})$ is the relative item popularity. We adopt the same power-law function as previous works [13, 15, 22] do in Equation 2, but set the exponent as a learnable parameter by $e = f_e(\mathbf{Q}_i)$. In essence, $a = f_a(\mathbf{Q}_i)$ aims to learn the relative popularity for unobserved negative interactions for items and $w = f_w(\mathbf{Q}_i)$ learns the trade-off between the popularity of positive and negative interactions. Moreover, we can view the propensity estimation in Equation 2 as a special case of our item-based propensity model when $w = 0$ and $e = \eta$.

3.3 Combinational Joint Learning Framework

Up to now, we have objective functions in Equation 1 for learning unbiased user-item relevance probability given we know the propensity (user-item exposure probability), and we also have the objective function in Equation 3 for learning unbiased propensity given we know the user-item relevance probability.

Naïve Joint Learning Method and Estimation-training Overlap Problem. Therefore, a straightforward idea is to combine them together in one model and jointly learn both of them. Concretely, assume we have a relevance model $\Psi = \{\mathbf{P}, \mathbf{Q}\}$ and a propensity model $\Phi = \{f_w, f_a, f_e\}$. For one observed user-item interaction $(u, i) \in \mathcal{D}$, we can first fix the propensity model Φ, and use the prediction $\widehat{O}_{u,i}$ of Φ as the propensity in Equation 1 to update the relevance model Ψ; then fix Ψ and update Φ based on Equation 3 with the prediction $\widehat{R}_{u,i}$ of Ψ as the relevance probability.

However, such a naïve method faces the *estimation-training overlap* problem. That is, the user-item pairs for training and propensity estimation (or relevance estimation) overlap with each other. More specifically, for a random user-item pair (u, i) in $\mathcal{D}$, we can use it to train the relevance model Ψ, and then use $\widehat{R}_{u,i}$ by Ψ as the relevance probability $\gamma_{u,i}$ in Equation 3 to update the propensity model Φ. This is problematic because Ψ has been trained by (u, i), so that $\gamma_{u,i} = \widehat{R}_{u,i}$ becomes the probability of u liking i given u has already provided positive feedback to i. Hence, $\gamma_{u,i}$ will be predicted as 1 by Ψ, which violates the definition of $\gamma_{u,i}$ as the probability parameter for the Bernoulli variable $R_{u,i}$. Similarly, using $\widehat{O}_{u,i}$ by Φ, which is trained by (u, i), as the propensity $\theta_{u,i}$ for updating Ψ brings the same problem.

Combinational Joint Learning. To address this problem, we propose a combinational joint learning framework, which separately learns unbiased relevance model Ψ and unbiased propensity model Φ by different data samples. The key idea is to split the training data into multiple chunks, and have multiple relevance sub-models and propensity sub-models so that the data chunks used for training any one of them and the chunks they predict relevance and propensity for do not overlap. Formally, we randomly divide the original training data $\mathcal{D}$ into C chunks with the same size: $\{\mathcal{D}_1, \ldots, \mathcal{D}_C\}$, C is a predefined combination hyper-parameter. Then, we define C relevance and propensity sub-models: $\{\Psi_1, \ldots, \Psi_C\}$ and $\{\Phi_1, \ldots, \Phi_C\}$. Each of the relevance sub-models and the propensity sub-models has the same structure as the conventional relevance and propensity model $\Psi = \{\mathbf{P}, \mathbf{Q}\}$ and $\Phi = \{f_w, f_a, f_e\}$. During training, for the c-th relevance and propensity sub-models Ψ_c and Φ_c, we will use all data chunks except $\mathcal{D}_c$ to update them, and output $\widehat{R}^c_{u,i}$ by Ψ_c and $\widehat{O}^c_{u,i}$ by Φ_c as the relevance probability $\gamma_{u,i}$ and propensity $\theta_{u,i}$ for user-item pairs in $\mathcal{D}_c$ for training other sub-models. And $\gamma_{u,i}$ and $\theta_{u,i}$ for chunks except $\mathcal{D}_c$ for training Ψ_c and Φ_c are provided by the other $C-1$ relevance and propensity sub-models. For example, Φ_1 and Ψ_1 are trained by $\{\mathcal{D}_2, \ldots, \mathcal{D}_C\}$ with $\gamma_{u,i}$ and $\theta_{u,i}$ provided by $\{\Phi_2, \ldots, \Phi_C\}$ and $\{\Psi_2, \ldots, \Psi_C\}$ correspondingly, and Φ_1 and Ψ_1 output $\theta_{u,i}$ and $\gamma_{u,i}$ for $\mathcal{D}_1$ for the training process of other sub-models. In this way, data for training and propensity estimation (or relevance estimation) does not overlap for all of the sub-models.

Yet, there is another issue. Each of the sub-models is only trained by partial training data ($C-1$ chunks), leading to information loss and compromised performance even if we average all sub-models as the final output. Hence, we further introduce a residual component to complement each sub-model. For example, for sub-models $\Psi_c = \{\mathbf{P}_c, \mathbf{Q}_c\}$ and $\Phi_c = \{f^c_w, f^c_a, f^c_e\}$ we have the residual components $\overline{\Psi}_c = \{\overline{\mathbf{P}}_c, \overline{\mathbf{Q}}_c\}$ and $\overline{\Phi}_c = \{\overline{f^c_w}, \overline{f^c_a}, \overline{f^c_e}\}$, and add the residual components to the original sub-models as the final models for output: $\Psi'_c = \{\mathbf{P}_c + \overline{\mathbf{P}}_c, \mathbf{Q}_c + \overline{\mathbf{Q}}_c\}$ and $\Phi'_c = \{f^c_w + \overline{f^c_w}, f^c_a + \overline{f^c_a}, f^c_e + \overline{f^c_e}\}$. The residual component is trained by the complete $\mathcal{D}$ with sub-models fixed, and the relevance and propensity are provided by all the sub-models. The training algorithm is shown in Algorithm 1. Last, after training, by averaging the output of $\{\Psi'_1 \ldots \Psi'_C\}$ and $\{\Phi'_1 \ldots \Phi'_C\}$, we have the final unbiased relevance and propensity predictions.

4 EXPERIMENTS

We conduct experiments on two real-world datasets to answer three research questions: **RQ1** How does the proposed method perform compared with state-of-the-art alternatives? **RQ2** How effective is the estimated propensity? and **RQ3** What is the impact of the combination hyper-parameter C and of the residual components in the proposed model?

4.1 Experimental Setup

Datasets. To evaluate unbiased recommendation, we need datasets with items uniformly exposed to users so that we can directly evaluate user-item relevance without influence of exposure. Thus, we use the Yahoo and Coat datasets, which are the only two publicly available datasets containing separate test sets where users provide feedback to uniformly drawn samples of items. **Yahoo! R3** (https://webscope.sandbox.yahoo.com/) contains over 300K ratings (1 to 5) from 15.4K users to 1K songs in the training set (a biased training set). Besides, an unbiased test set is collected by sampling a subset of 5.4K users, each of whom is randomly assigned 10 songs, and asked to provide ratings to these random items. Following the preprocessing procedure in [22], we regard ratings ≥ 4 as positive feedback, and we randomly split 10% of the training set to be a biased validation set. **Coat Shopping** [16] contains around 7K ratings (1 to 5) from 290 users to 300 coats in the training set (a biased training set). Similar to the Yahoo dataset, the Coat dataset also has an unbiased random test set by asking all 290 users to provide ratings to 16 randomly selected coats.

Metrics. Following [13, 15], we adopt three widely used implicit recommendation evaluation metrics – *DCG@k* (Discounted Cumulative Gain) and *MAP@k* (Mean Average Precision). The detailed formulations can be found in [13, 15]. We report results with $k = 1, 2, 3$ since the number of candidate items for ranking is small in the test set. We rank the 10 exposed items for each user in the Yahoo dataset, and 16 items in the Coat dataset.

Combinational Approaches. We evaluate combinational joint learning for both a point-wise version (**CJMF**) and a pair-wise variation (**CJBPR**). CJBPR uses the unbiased Bayesian Personalized Ranking (BPR) loss proposed by [13] for

Table 1. Recommendation performance comparison, where best baselines are marked by underlines.

			Point-wise models							Pair-wise models			
			MF -RMSE	MF -CE	RelMF -RMSE	RelMF -CE	NJMF	CJMF	Δ	BPR	UBPR	CJBPR	Δ
Yahoo	DCG	@1	0.5314	0.5275	0.5364	0.5339	0.5403	**0.5610**	4.58%	0.5409	0.5433	**0.5648**	3.96%
		@2	0.7297	0.7385	0.7353	0.7398	0.7434	**0.7746**	4.71%	0.7451	0.7493	**0.7750**	3.42%
		@3	0.8520	0.8582	0.8595	0.8616	0.8678	**0.8960**	4.00%	0.8672	0.8777	**0.8972**	2.22%
	MAP	@1	0.5314	0.5275	0.5364	0.5339	0.5403	**0.5610**	4.58%	0.5419	0.5433	**0.5648**	3.96%
		@2	0.6189	0.6178	0.6203	0.6220	0.6256	**0.6475**	4.09%	0.6263	0.6295	**0.6496**	3.19%
		@3	0.6420	0.6419	0.6433	0.6465	0.6486	**0.6694**	3.54%	0.6491	0.6532	**0.6721**	2.88%
Coat	DCG	@1	0.5305	0.5485	0.5485	0.5612	0.5696	**0.5907**	5.26%	0.5316	0.5738	**0.5907**	2.94%
		@2	0.7608	0.7695	0.7881	0.7848	0.7949	**0.8223**	4.34%	0.7739	0.7868	**0.8223**	4.51%
		@3	0.9190	0.9298	0.9337	0.9367	0.9431	**0.9679**	3.33%	0.9300	0.9387	**0.9595**	2.21%
	MAP	@1	0.5305	0.5485	0.5485	0.5612	0.5696	**0.5907**	5.26%	0.5316	0.5738	**0.5907**	2.94%
		@2	0.6118	0.6203	0.6371	0.6435	0.6477	**0.6709**	4.26%	0.6181	0.6392	**0.6709**	4.95%
		@3	0.6255	0.6399	0.6498	0.6494	0.6572	**0.6741**	3.73%	0.6378	0.6596	**0.6818**	3.36%

the relevance model, and uses the same propensity loss in Equation 3 for the propensity estimation. Since the output of the relevance model $\widehat{R}_{u,i}$ in CJBPR is not a probability, for a user u, we further use a softmax function on $\widehat{R}_{u,*}$ to have a multinomial distribution for predicted scores and then divide the scores by the maximum score in the multinomial distribution to transform scores to relevance probabilities $\gamma_{u,*}$ in Equation 3.

Baselines. For fair comparison, we compare CJMF with point-wise baselines, and compare CJBPR with pair-wise baselines. For point-wise baselines, we use the following biased models: **MF-RMSE** [8], the most commonly used matrix factorization model for implicit RecSys with RMSE loss; **MF-CE**, a variation of MF-RMSE adopting cross entropy loss (to have a fair comparison with CJMF that also uses cross entropy loss). We also consider the following unbiased baselines: **RelMF-RMSE** [15], an unbiased model adopting the IPS approach and the RMSE loss; **RelMF-CE** is a variation of RelMF-RMSE which adopts cross entropy loss to have a fair comparison with the proposed CJMF. For pairwise baselines, we adopt the biased **BPR** [12] model and **UBPR** [13], an unbiased version of BPR which also uses the IPS approach.

All point-wise and pair-wise unbiased baselines [13, 15] use the power-law function of item popularity introduced in Equation 2 with $\eta = 0.5$ (the same as in the original papers [13, 15]) as propensity. We also adopt the propensity clipping approach [13, 15] to reduce the variance for all unbiased baselines and also our proposed models.

Reproducibility. We implement the proposed models based on Tensorflow [1] and use Adam [7] optimizer. We set the learning rate as 0.001, the batch size as 1024, the latent dimension as 100, and the negative sampling rate as 5 for all models and datasets. For other hyper-parameters, we tune them by grid search on the biased validation sets with the self-normalized inverse propensity scoring (SNIPS) estimator [22] as the performance indicator. More specifically, we set $C = 8$ for both Yahoo and Coat datasets for CJMF and set $C = 6$ for CJBPR. Since the proposed CJMF and CJBPR adopt the combinational joint learning method leading to more model parameters than baselines, to have a fair comparison, we run point-wise baselines for 16 times (run pair-wise baselines for 12 times) and average their outputs as final predictions. Code, data, and experimental settings are at https://github.com/Zziwei/Unbiased-Propensity-and-Recommendation.

4.2 RQ1: Comparing Recommendation Performance

We begin by investigating the recommendation performance of the proposed CJMF and CJBPR approaches compared with corresponding point-wise and pair-wise baselines. Detailed results for all models are shown in Table 1. First, comparing among point-wise models and among pair-wise models, we observe that the proposed CJMF and CJBPR can significantly outperform corresponding baselines (the best baselines are marked by underlines in Table1), which indicates that the proposed combinational joint learning approach is effective for unbiased implicit recommendation.

Second, we also implement the naïve joint learning model (denoted as NJMF in Table 1) introduced in Section 3.3 which has the estimation-training overlap issue. From the table we observe that CJMF produces better recommendation results than NJMF, demonstrating the effectiveness of the proposed combinational joint strategy over the naïve approach. Last, by comparing CJMF and CJBPR, we find that results of CJBPR are better than CJMF but the difference is small, which is not as obvious as the difference between conventional MF and BPR. Since the propensity and the relevance are modeled differently in CJBPR (the propensity is modeled as probabilities while relevance is modeled as real-number scores), we see some limits to the effectiveness of the joint learning. We leave further study along this line to future work.

4.3 RQ2: Investigating the Effectiveness of Estimated Propensity

Next, we study the effectiveness of the estimated propensity by comparing the performance of three unbiased baselines using Equation 2 as propensity estimation versus using $\widehat{O}_{u,i}$ predicted by CJMF and CJBPR as propensity estimation. Results presented in Figure 1 show that for both datasets, the two variations of RelMF can perform better if use the estimated propensity from CJMF. The same conclusion can be drawn that with the estimated propensity from CJBPR, UBPR can perform better.

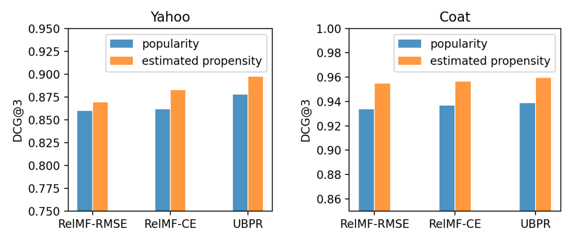

Fig. 1. Comparing unbiased models with item popularity as propensity and with estimated propensity from proposed models.

4.4 RQ3: Investigating the Impact of Hyper-parameter C and Residual Component

Finally, we investigate the impact of the combination hyper-parameter C on CJMF. We vary C from 2 to 16 and show the results of CJMF as the red line in Figure 2. We see that the performance of CJMF improves rapidly then converges as C increases, reaching a peak level when $C \geq 5$.

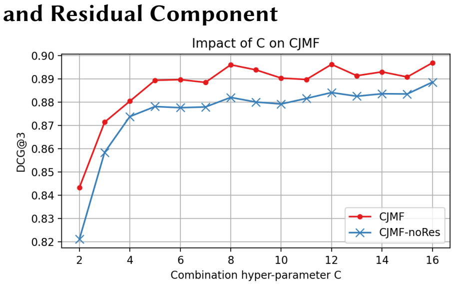

Fig. 2. *DCG@3* of CJMF and CJMF without residual components on the Yahoo dataset, with varying C.

Then, we study the effect of the residual components. We denote the variation of CJMF without the residual component as CJMF-noRes, which directly averages the output of all sub-models as the final prediction of the complete model. The $DCG@3$ results of CJMF-noRes are plotted in Figure 2 as the blue lines. We observe the effectiveness of the residual components by comparing with the complete model (the red lines in the figure). Note that CJBPR has similar patterns as CJMF demonstrated in Figure 2.

5 CONCLUSION AND FUTURE WORK

In this paper, we propose a combinational joint learning framework, which jointly learns unbiased relevance and propensity simultaneously, to produce unbiased recommendations based on biased implicit data. Extensive experiments on two public datasets show the effectiveness of the proposed method. There are two lines of future work: i) investigate how to more effectively adapt the combinational joint learning framework to pair-wise algorithms; and ii) study how to design user-item based propensity models to connect propensity estimation with both users and items.

ACKNOWLEDGMENTS

This work is, in part, supported by NSF #IIS-1841138 and DARPA W911NF-16-1-0565.

REFERENCES

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: a system for large-scale machine learning.. In *OSDI*, Vol. 16. 265–283.

[2] Stephen Bonner and Flavian Vasile. 2018. Causal embeddings for recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 104–112.

[3] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*. 173–182.

[4] José Miguel Hernández-Lobato, Neil Houlsby, and Zoubin Ghahramani. 2014. Probabilistic matrix factorization with non-random missing data. In *International Conference on Machine Learning*. 1512–1520.

[5] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE International Conference on Data Mining*. Ieee, 263–272.

[6] Santosh Kabbur, Xia Ning, and George Karypis. 2013. Fism: factored item similarity models for top-n recommender systems. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. 659–667.

[7] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[8] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[9] Dawen Liang, Laurent Charlin, and David M Blei. 2016. Causal inference for recommendation. In *Causation: Foundation to Application, Workshop at UAI*. AUAI.

[10] Benjamin M Marlin and Richard S Zemel. 2009. Collaborative prediction and ranking with non-random missing data. In *Proceedings of the third ACM conference on Recommender systems*. 5–12.

[11] Andriy Mnih and Russ R Salakhutdinov. 2008. Probabilistic matrix factorization. In *Advances in neural information processing systems*. 1257–1264.

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

[13] Yuta Saito. 2019. Unbiased Pairwise Learning from Implicit Feedback. In *NeurIPS 2019 Workshop on Causal Machine Learning*.

[14] Yuta Saito. 2020. Asymmetric Tri-training for Debiasing Missing-Not-At-Random Explicit Feedback. In *The 43st International ACM SIGIR Conference on Research & Development in Information Retrieval*.

[15] Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. 2020. Unbiased Recommender Learning from Missing-Not-At-Random Implicit Feedback. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 501–509.

[16] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. *arXiv preprint arXiv:1602.05352* (2016).

[17] Harald Steck. 2010. Training and testing of recommender systems on data missing not at random. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. 713–722.

[18] Harald Steck. 2011. Item popularity and recommendation accuracy. In *Proceedings of the fifth ACM conference on Recommender systems*. 125–132.

[19] Harald Steck. 2013. Evaluation of recommendations: rating-prediction and ranking. In *Proceedings of the 7th ACM conference on Recommender systems*. 213–220.

[20] Xiaojie Wang, Rui Zhang, Yu Sun, and Jianzhong Qi. 2019. Doubly robust joint learning for recommendation on data missing not at random. In *International Conference on Machine Learning*. 6638–6647.

[21] Yixin Wang, Dawen Liang, Laurent Charlin, and David M Blei. 2018. The deconfounded recommender: A causal inference approach to recommendation. *arXiv preprint arXiv:1808.06581* (2018).

[22] Longqi Yang, Yin Cui, Yuan Xuan, Chenyang Wang, Serge Belongie, and Deborah Estrin. 2018. Unbiased offline recommender evaluation for missing-not-at-random implicit feedback. In *Proceedings of the 12th ACM Conference on Recommender Systems*. 279–287.

[23] Xing Zhao, Ziwei Zhu, Yin Zhang, and James Caverlee. 2020. Improving the Estimation of Tail Ratings in Recommender System with Multi-Latent Representations. In *Proceedings of the 13th International Conference on Web Search and Data Mining*. 762–770.