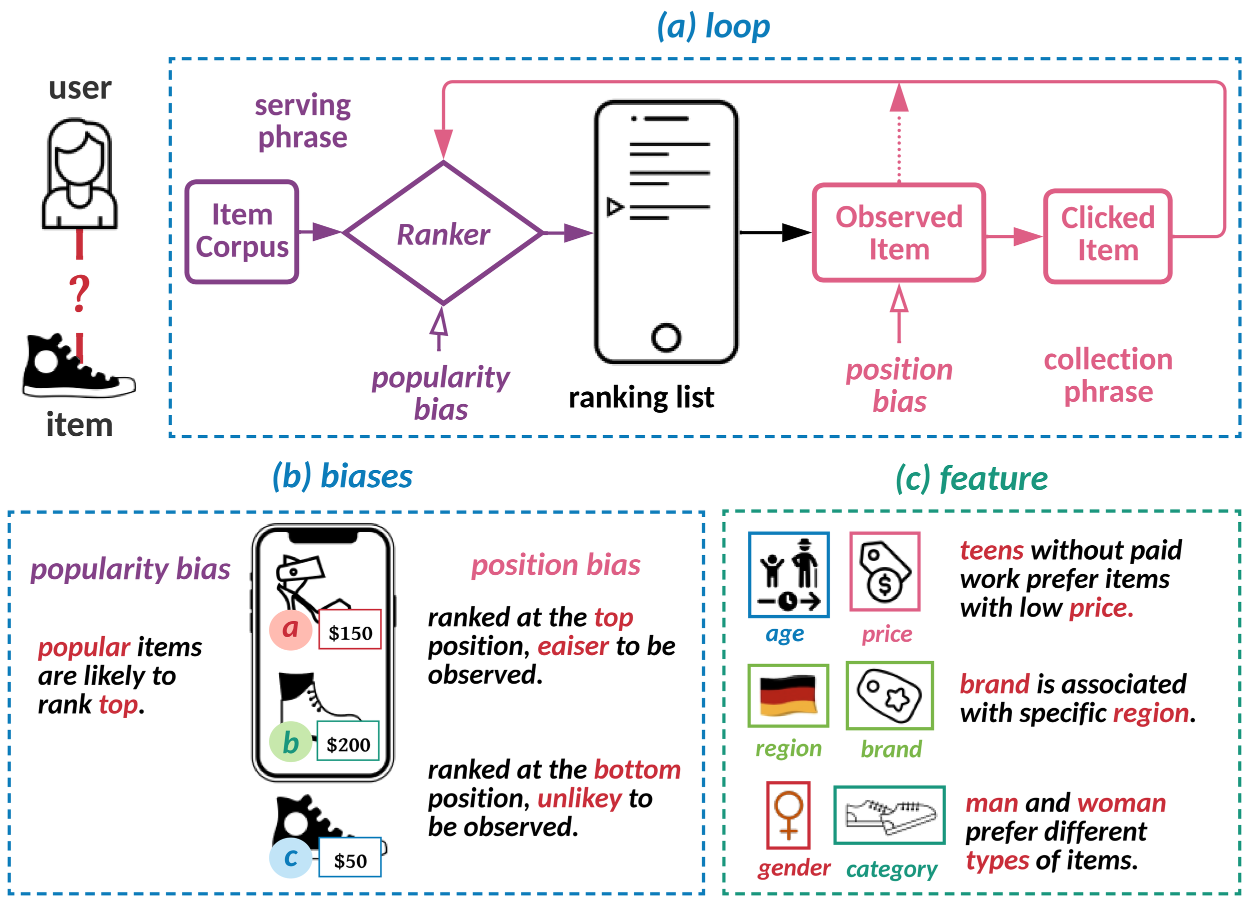
Research Track | Search

InfoRank: Unbiased Learning-to-Rank via Conditional Mutual Information Minimization

Jiarui Jin[1], Zexue He[2], Mengyue Yang[3], Weinan Zhang[1], Yong Yu[1], Jun Wang[3], Julian McAuley[2]

[1]Shanghai Jiao Tong University, [2]University of California San Diego, [3]University College London

Background: Biases in Learning-to-Rank



An illustrated example of the feedback loop, position bias, and popularity bias in learning-to-rank. Within this process, the ranking system blends user and item features (c) with implicit feedback to generate the final ranking list. However, this system is susceptible to both position bias and popularity bias (b). Furthermore, these biases tend to be amplified within the feedback loop (a), potentially resulting in a "rich-get-richer" dilemma.

Position bias: items occupying higher positions are more prone to being both observed and subsequently clicked. Consequently, training a ranker directly on click data may lead to it primarily estimating the position order rather than the personalized relevance of items.

$$P(C=1|X=\mathbf{x}) = P(R=1|X=\mathbf{x}) \cdot P(O=1|X=\mathbf{x}).$$

Popularity bias: items with higher levels of popularity are more likely to be posted and then are more frequently observed and clicked. Consequently, optimizing a ranker's performance directly on click data may result in it primarily estimating the popularity order rather than personalized relevance.

$$P(R=1|C=\{c=1\}_d, X=\mathbf{x}) = P(R=1|\mathcal{R}=\{r=1\}_d, X=\mathbf{x}).$$

Causality in Learning-to-Rank

Our main idea is to consolidate the impacts of those biases into a single observation factor, thereby providing a unified approach to addressing bias-related issues.

We consider the observation factor as the "sensitive attribute". In this regard, an ideal ranker should adhere to the following principle: for any user u and item d, given their associated feature vector $\mathbf{x}$, we have:

$$P(R=r|O=1, X=\mathbf{x}) = P(R=r|O=0, X=\mathbf{x})$$

holds for any relevance score $r \in \{0,1\}$, and any observation value $o \in \{0,1\}$ attainable by O.

Position bias: in the previous equation, the estimation of an item's relevance can still be affected by whether it has been observed or not. Therefore, we advocate for an additional step to ensure the conditional independence between R and O

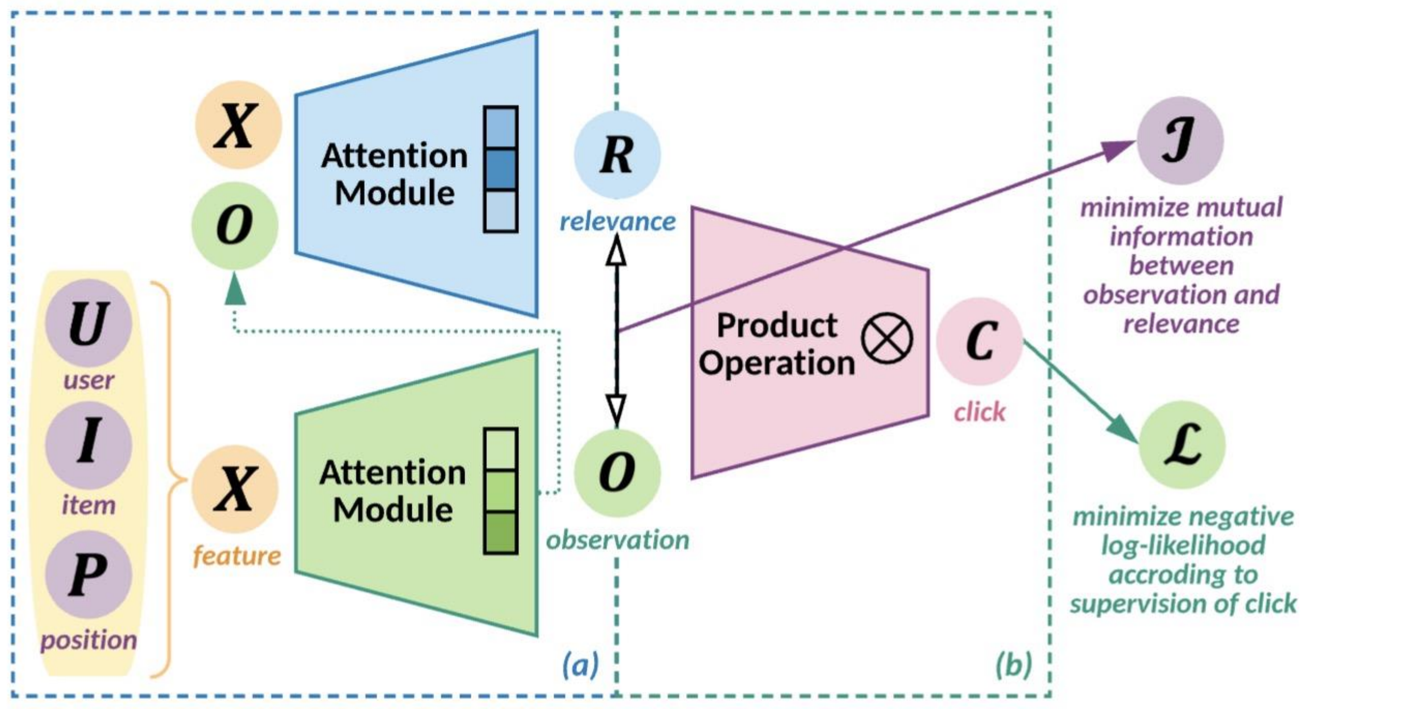
$$P(R=1|X=\mathbf{x}) = P(R=1|O=o, X=\mathbf{x}),$$

Popularity bias: Consider that given the features of an item d, its previous clicks (i.e., $\{c=1\}_d$) only occur when d is both relevant (i.e., $\{r=1\}_d$) and observed (i.e., $\{o=1\}_d$) by users. Following this, we can proceed to derive:

$$P(R=1|\mathcal{C}, X) = \frac{P(R=1|\mathcal{O}, X)}{P(R=1|X)} P(R=1|\mathcal{R}, X).$$

Observing $\mathcal{O}=\{o=1\}_d$ and $O=1$ are closely correlated, given that they both signify user observations, we argue that reinforcing $O=1$ and $R=1$'s independence conditioned on $X=\mathbf{x}$ can lead to an approximation where $P(R=1|\mathcal{O}=\{o=1\}_d, X=\mathbf{x})/P(R=1|X=\mathbf{x})$ approaches 1. The remaining part $P(R=1|\mathcal{R}=\{r=1\}_d, X=\mathbf{x})$ reflects the ranker's inductive capacity. This capacity corresponds to the process of learning from the historical records $\mathcal{R}=\{r=1\}_d$ to infer $R=1$, specifically utilizing the past relevance feedback for item d to infer current behaviour regarding d.

InfoRank: Overall Architecture



We first leverage an attention mechanism to mine correlations between user-item features, as shown in (a); and we then introduce a regularization formulation (i.e., $\mathcal{I}$) aimed at establishing conditional mutual information to ensure that relevance becomes conditionally independent of the observation factor, as shown in (b). To capture relevance within biased feedback, we incorporate this regularization term with supervision (i.e., $\mathcal{L}$) over user behaviours.

We note that InfoRank remains working even in scenarios where there is no observation information available within user browsing logs. In such cases, we substitute real observations with estimated ones.

Conditional Mutual Information

$\mathcal{I}$ is defined as

$$\mathcal{I} := \mathcal{I}(R;O|X) = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\left[\mathcal{I}(R;O|X=\mathbf{x})\right],$$

where

$$\begin{aligned}
&\mathcal{I}(R;O|X=\mathbf{x}) \\
&= \sum_{R,O} P(R,O|X=\mathbf{x}) \cdot \ln \frac{P(R,O|X=\mathbf{x})}{P(R|X=\mathbf{x}) \cdot P(O|X=\mathbf{x})} \\
&= \sum_{R,O} P(R|O,X=\mathbf{x}) \cdot P(O|X=\mathbf{x}) \cdot \ln \frac{P(R|O,X=\mathbf{x})}{P(R|X=\mathbf{x})}.
\end{aligned}$$

We can define $\Delta\text{CI} := |P(R=1|O=1,X=\mathbf{x}) - P(R=1|O=0,X=\mathbf{x})|$.

Proposition 3.1. *Given that relevance, click, and observation variables are binary (i.e., $R,C,O \in \{1,0\}$), for any user-item pair with feature X, the following statements are equivalent:*

- *The relevance R and observation O are conditionally independent given X. In other words, $P(R,O|X) = P(R|X) \cdot P(O|X)$. That is, $P(R|O=1,X) - P(R|O=0,X) = 0$.*
- *The conditional mutual information between relevance R and observation O (later defined in Eq. (13)) is zero, i.e., $\mathcal{I}(R;O|X) = 0$.*
- *The conditional independence score ΔCI is zero.*

Optimization Functions

Regarding $\mathcal{L}$, given that click signals are binary, we employ Binary Cross Entropy (BCE) loss for click supervision. The BCE loss can be formulated as:

$$\mathcal{L} = -\sum_{(c,\mathbf{x})\in\mathcal{D}} \Big(c \cdot \log P(\hat{c}|\mathbf{x}) + (1-c) \cdot \log(1 - P(\hat{c}|\mathbf{x}))\Big),$$

$$\arg\min_{\theta} \mathcal{L} + \eta \cdot \mathcal{I},$$

Table 2: Comparison of different unbiased learning-to-rank methods on Yahoo Search Engine, LETOR Webpage Ranking, and Adressa Recommender System datasets. UBM is used as a click generation model. * indicates $p < 0.001$ in significance tests compared to the best baseline.

Ranker	Debiasing Method	Yahoo (UBM)				LETOR (UBM)				Adressa (UBM)			
		MAP	N@3	N@5	N@10	MAP	N@3	N@5	N@10	MAP	N@3	N@5	N@10
InfoRank (Ranking)	Labeled Data	.856	.755	.760	.795	.695	.381	.468	.563	.821	.714	.727	.754
	InfoRank (Debiasing)	**.845***	**.736***	**.739***	**.779***	**.650***	**.380***	**.460***	**.541***	**.801***	**.691***	**.715***	**.739***
	Regression-EM	.837	.683	.692	.731	.634	.374	.442	.535	.794	.673	.706	.731
	Randomization	.835	.680	.689	.728	.630	.368	.437	.515	.792	.668	.695	.728
	Click Data	.823	.670	.678	.720	.622	.356	.428	.489	.782	.648	.677	.707
LambdaMART	Labeled Data	.854	.745	.757	.790	.685	.380	.461	.558	.814	.709	.722	.747
	Ratio-Debiasing	.832	.712	.722	.755	.631	.365	.421	.506	.791	.669	.702	.730
	Regression-EM	.827	.680	.693	.741	.628	.356	.411	.490	.785	.650	.681	.711
	Randomization	.824	.675	.687	.725	.624	.346	.407	.482	.784	.648	.678	.705
	Click Data	.814	.666	.673	.712	.614	.339	.396	.473	.779	.635	.664	.694
DNN	Labeled Data	.831	.685	.705	.737	.678	.364	.454	.551	.802	.700	.722	.745
	InfoRank (Debiasing)	.828	.683	.696	.734	.637	.360	.416	.499	.786	.667	.692	.725
	Dual Learning	.825	.680	.693	.730	.625	.352	.410	.487	.784	.663	.688	.720
	Regression-EM	.823	.676	.689	.726	.618	.347	.400	.479	.779	.656	.675	.713
	Randomization	.822	.677	.686	.724	.617	.346	.397	.477	.777	.644	.664	.701
	Click Data	.817	.665	.671	.710	.612	.335	.387	.469	.775	.633	.659	.688

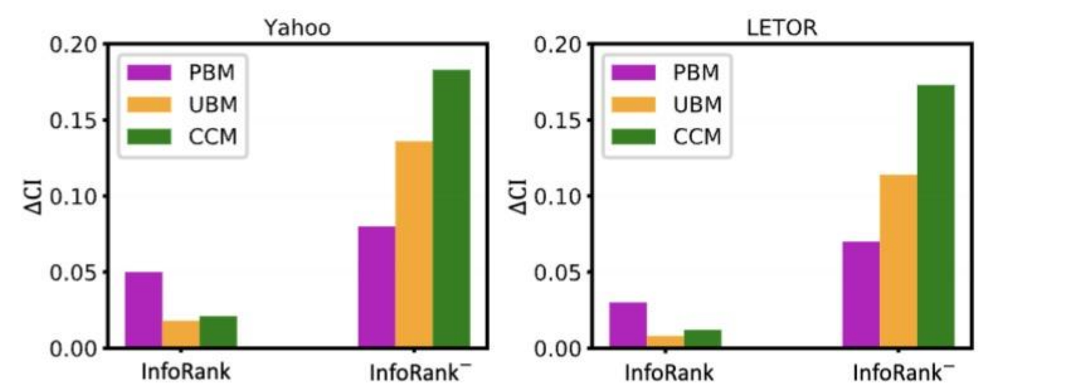


Figure 4: Comparison of InfoRank and InfoRank⁻ under different click generation models and datasets in terms of the ΔCI metric.

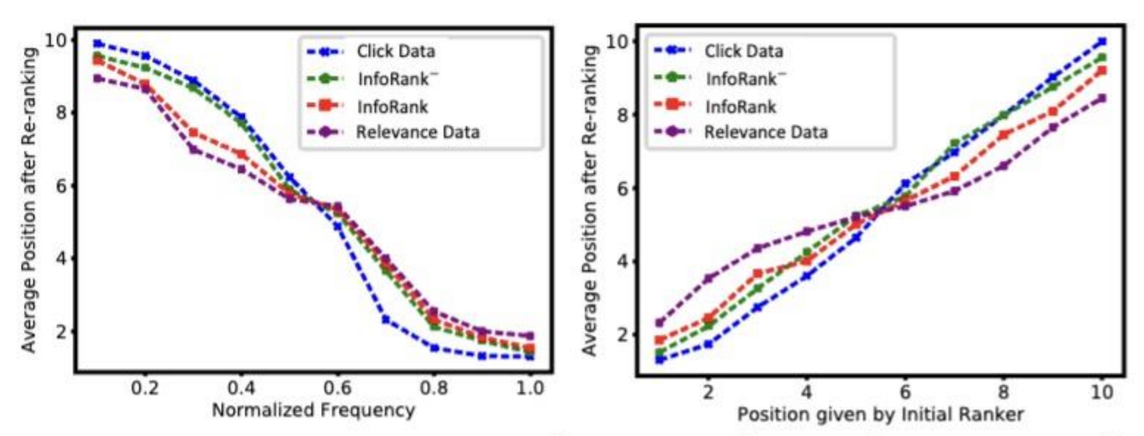


Figure 3: Average positions after re-ranking of items at each normalized frequency (in the left subfigure); or at each original position (in the right subfigure) by different debiasing methods together with InfoRank and InfoRank⁻ on Yahoo.