

DebiasRec: Bias-aware User Modeling and Click Prediction for Personalized News Recommendation

Jingwei Yi
University of Science and Technology
of China
yjiw1029@mail.ustc.edu.cn

Fangzhao Wu
Microsoft
wufangzhao@gmail.com

Chuhan Wu
Tsinghua University
wuchuhan15@gmail.com

Qifei Li
Beihang University
wxliqifei@sina.com

Guangzhong Sun
University of Science and Technology
of China
gzsun@ustc.edu.cn

Xing Xie
Microsoft
xingx@microsoft.com

ABSTRACT

News recommendation is critical for personalized news access. Existing news recommendation methods usually infer users' personal interest based on their historical clicked news, and train the news recommendation models by predicting future news clicks. A core assumption behind these methods is that news click behaviors can indicate user interest. However, in practical scenarios, beyond the relevance between user interest and news content, the news click behaviors may also be affected by other factors, such as the bias of news presentation in the online platform. For example, news with higher positions and larger sizes are usually more likely to be clicked. The bias of clicked news may bring noises to user interest modeling and model training, which may hurt the performance of the news recommendation model.

In this paper, we propose a bias-aware personalized news recommendation method named DebiasRec, which can handle the bias information for more accurate user interest inference and model training. The core of our method includes a bias representation module, a bias-aware user modeling module, and a bias-aware click prediction module. The bias representation module is used to model different kinds of news bias and their interactions to capture their joint effect on click behaviors. The bias-aware user modeling module aims to infer users' debiased interest from the clicked news articles by using their bias information to calibrate the interest model. The bias-aware click prediction module is used to train a debiased news recommendation model from the biased click behaviors, where the click score is decomposed into a preference score indicating user's interest in the news content and a news bias score inferred from its different bias features. Experiments on two real-world datasets show that our method can effectively improve the performance of news recommendation.

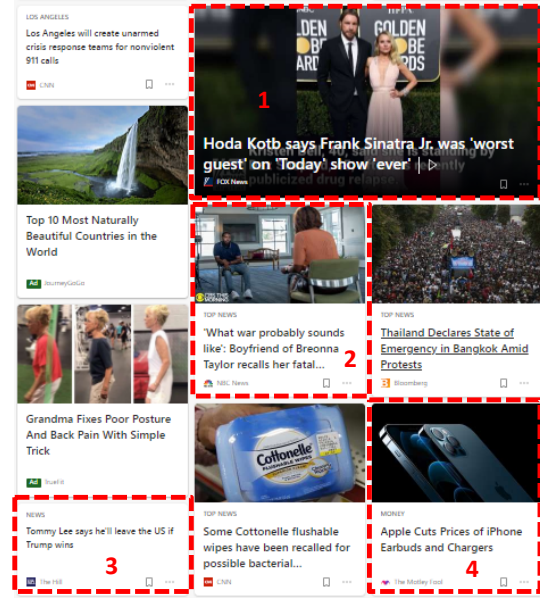


Figure 1: Illustration of different news bias (e.g., position and size) in an online news platform.

KEYWORDS

News Recommendation, User Modeling, Bias

ACM Reference Format:

Jingwei Yi, Fangzhao Wu, Chuhan Wu, Qifei Li, Guangzhong Sun, and Xing Xie. 2021. DebiasRec: Bias-aware User Modeling and Click Prediction for Personalized News Recommendation. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

With the development of the World Wide Web, people's reading habits have shifted from paper-based reading to Internet reading [20, 34]. News websites such as Google News ¹, Toutiao ²,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
Conference'17, July 2017, Washington, DC, USA

© 2021 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

¹<https://news.google.com/>

²<https://www.toutiao.com/>

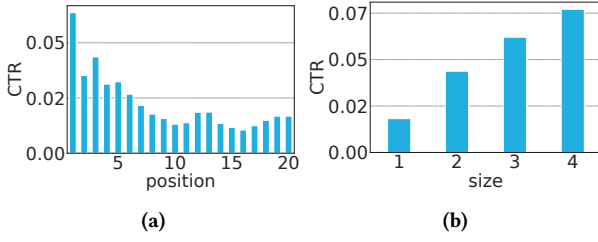


Figure 2: Overall CTR of news at different positions and in different sizes in a commercial online news platform.

and MSN News³ which collect and display massive news from various sources have attracted many users [7, 41]. One vital problem for these news websites is that the large quantities of news generated every day may overwhelm users [23]. Thus, it is important to utilize personalized news recommendation methods to help users find the news they are interested in and reduce their information overload [23, 39].

News recommendation has been extensively studied in recent years [2, 13, 17, 21, 30, 33]. These methods usually infer user interests from the content of their clicked news articles. For example, Wang et al. [34] used a knowledge-aware CNN to learn news representations. A candidate-aware attention network is utilized to learn user representations from clicked news. Wu et al. [39] proposed to learn news content representations from news titles via multi-head self-attention, and they learned user interest representations from their clicked news articles via multi-head self-attention. A core assumption behind these methods is that news click behaviors can indicate user interest. However, users' news click behaviors may also be affected by many other factors, such as position bias [9, 10] and size bias [29]. For example, as shown in Figure 1, the second and the forth news are displayed at different positions, while the first and the third news are displayed in different sizes. As discussed in [28], the click-through rate (CTR) declines rapidly with lower displayed positions. The same phenomenon is observed in the news dataset as well. As shown in Figure 2, the CTR drops with lower displayed positions and increases with larger displayed sizes. These observations imply that biases may have a strong impact on user click behaviors because users may click the "attractive" news instead of the news they are really interested in [35]. The bias in news clicks may bring noise to user interest modeling and model training, which may hurt the model performance. Therefore, it is critical to reduce the influence of biases in the click data for better targeting user interests. However, existing news recommendation methods do not consider the bias effect in user modeling and model training, which may be suboptimal.

In this paper, we propose a bias-aware personalized news recommendation method named **DebiasRec**⁴, which can handle the bias information in click behaviors for both user interest inference and recommendation model training. Our DebiasRec method contains three core modules, i.e., the bias representation model (BRM), the bias-aware user modeling (BAUM) and the bias-aware click prediction (BACP). The bias representation module BRM is used

to learn bias representations by incorporating different kinds of bias features and their interactions to model their joint effect on click behaviors. The bias-aware user modeling module BAUM is used to infer unbiased user interest from the biased historical news clicked behaviors. In this module, we propose a bias-aware attention network to select informative clicked news for user interest modeling, which can distinguish whether the user is actually interested in the news content or just attracted by the presentation bias. In the bias-aware click prediction module BACP, we propose to decompose the click prediction score into a bias score and a preference score. The bias score is inferred from the bias features of candidate news to model their effect on click. The preference score is used to measure user's personalized interest in the content of candidate news. We only use the preference scores for candidate news ranking in online recommendation. Extensive experiments on two real-world datasets show that DebiasRec can effectively improve the performance of news recommendation by handling the bias in user modeling and model training.

The main contributions of our work are summarized as follows:

- We propose a bias-aware news recommendation method which can handle the bias information in click behaviors for accurate user interest inference and model training.
- We propose a bias representation model to jointly model different bias features and their interactions.
- We propose a bias-aware user modeling method to infer accurate user interest from the biased click behaviors, and a bias-aware click prediction method for training debiased news recommendation models.
- We conduct experiments on real-world datasets to verify the effectiveness of the proposed methods.

2 RELATED WORKS

2.1 News Recommendation

Personalized news recommendation is a core technique on online news platforms to improve users' reading experience [23]. Accurate user interest modeling and click prediction are critical for news recommendation [43]. Existing news recommendation methods usually model user interests from their clicked news and predict click scores based on the relevance between user and candidate news representations [1, 13, 23, 40, 42, 46]. For example, Wang et al. [34] proposed to model user interests from clicked news based on their relevance to candidate news, and predict click scores based on candidate news and user embeddings via a feed-forward neural network. Wu et al. [39] proposed to use an attention network to select clicked news for learning user interest representations, and they used the inner product between candidate news and user representations to predict click scores. Wu et al. [41] proposed to use multi-head self-attention networks to model the relatedness between clicked news, and they used an attention network to select informative news clicks for learning user representations. The click scores are also computed by the inner product between user and news embeddings. These news recommendation methods assume that the click behaviors of users can indicate user interests [33]. However, users may click news not only due to the interest in the news title, but also the attraction of several biases such as large sizes and top positions [4, 6, 35]. The bias information encoded

³<https://www.msn.com/news>

⁴Our source codes will be soon released.

in news click behaviors will propagate to user interest modeling and model training, which is harmful for modeling performance and user experience. Different from these methods, our approach can handle and alleviate the bias information for both user interest inference and recommendation model training.

2.2 Bias-aware CTR Prediction

Click logs usually provide implicit but abundant user feedback, thereby they are widely used for modeling user interests and training recommendation models [38, 44, 45]. However, click logs are often biased and cannot fully reflect user preferences [9, 10, 24]. Thus, it is important to eliminate the bias effect in click data. Several early methods for click data debiasing are based on probabilistic click models [3, 8], which are initially proposed to solve the click bias problem in web search scenario for calculating the real relevance between query and document in web search. For example, Craswell et al. [6] propose a cascade model, which is one of the earliest click modeling methods. They assumed that users examine the documents sequentially and stop when the first click happens, and they used a Bayesian Network to calculate the unbiased relevance score. However, the cascade model is not suitable for modeling the effect of other kinds of bias like vertical bias, where items are not displayed in a sequential way. Based on the cascade hypothesis, Chen et al. [4] and Wang et al. [32] further proposed click models which can handle both position bias and vertical bias. However, these methods assume that the same pair of query and user appears multiple times in the training set, which is not suitable for the recommendation scenario because users usually do not click the same item repeatedly.

Several methods explored to treat the bias as a counterfactual effect for learning unbiased learning-to-rank systems [15, 36]. They usually rely on Inverse Propensity Weighting (IPW) to estimate the propensity weight with randomization. However, displaying news randomly will affect user reading experience and it is not suitable for these scenarios lack of random dataset. To resolve the problem, Wang et al. [37] proposed to apply a regression-based EM approach to estimate the propensity in unbiased learning-to-rank frameworks. However, these methods do not consider the bias effect in user interest modeling.

Several other methods employ position models to reduce the effect of position bias in recommendation [6]. Their basic idea is incorporating position information of items in the model training stages to model its effect, and deactivate it in the test stage. For example, Ling et al. [19] proposed to use the item position as the model input in the training stage, and replace it with a default position number in the inference stage. However, the ranking results may be different when inputting different default positions. Guo et al. [12] proposed to model the click probability as the multiplication of the relevance score and the probability that a user observes the item. They used a position module to compute observation probability based on positions and remove the position module in the online inference stage. These methods only consider a single kind of bias, i.e., position bias, and they are not able to eliminate the joint effect of multiple kinds of bias. Different from them, our approach can model the effect of multiple kinds of bias and meanwhile consider

their interaction, which can help better eliminate bias information in user interest modeling and model training.

3 PROBLEM FORMULATION

In this section, we present a formal definition of the problem studied in this paper. The task of our news recommendation model is predicting whether a user u will click on a candidate news n_c based on the content information of candidate news and the user interests inferred from clicked news. Each news is represented by its title texts. In addition, we assume the user u has M clicked news, which is denoted as $[n_1, n_2, \dots, n_M]$ (sorted by their click time). Each historical click n_i is associated with a position p_i and a size s_i for display. The recommendation model needs to compute a click score s_c for each pair of user and candidate news (u, n_c) , which is further used to rank candidate news according to user interests. Since user click behaviors are heavily affected by many bias information such as the position and size of news, both the user interests inferred from click behaviors and the click score for news ranking may also be biased. Thus, the goal of our approach is eliminate the bias information in user interest modeling and click prediction to help better target user interests for news recommendation.

4 METHODOLOGY

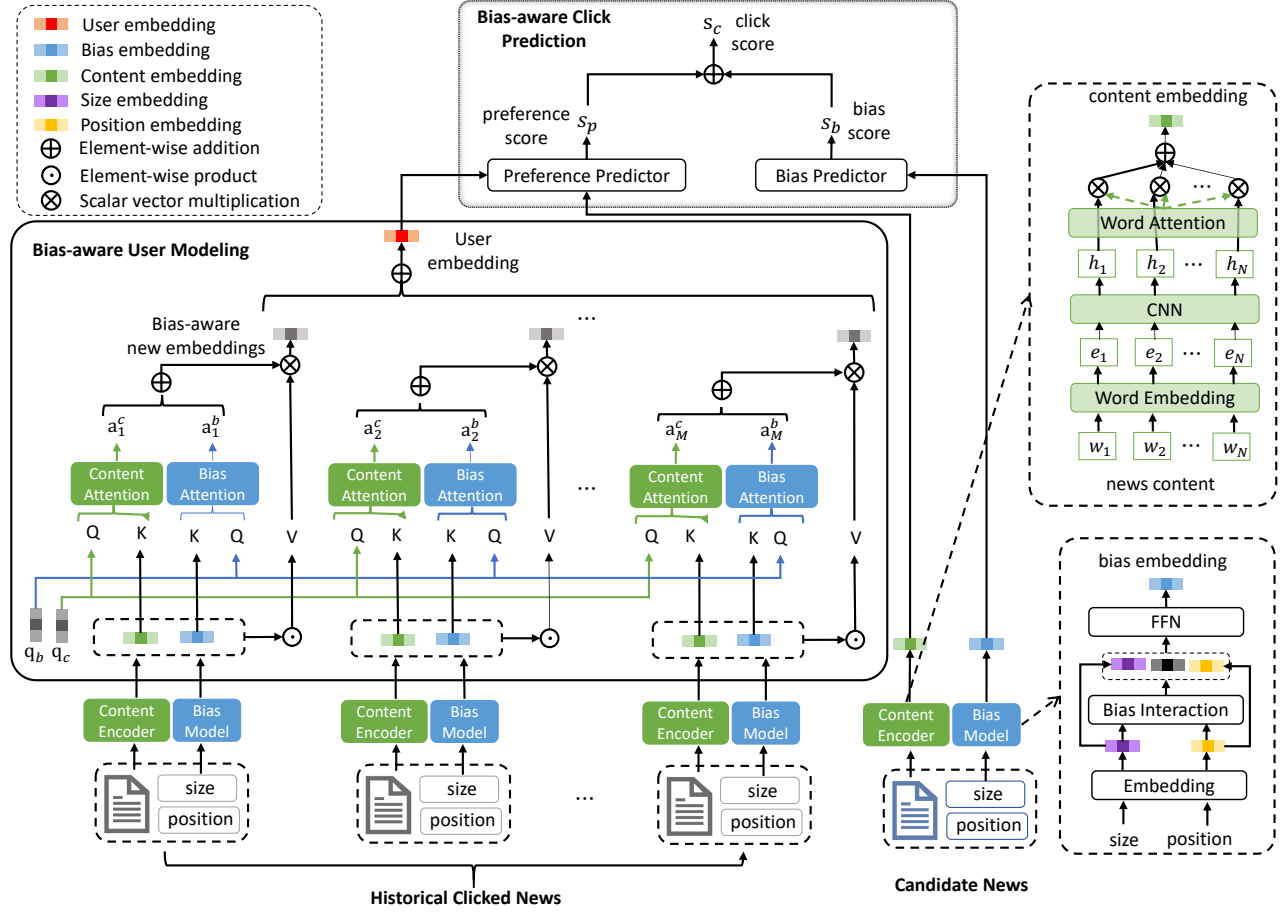
In this section, we introduce our bias-aware user modeling and click prediction method for news recommendation (DebiasRec). It considers multiple kinds of biases in news clicks and can reduce their effects on user interest modeling and click prediction for better targeting user interests. Figure 3 shows the overall framework of our DebiasRec model. Our DebiasRec approach contains four modules, i.e. content encoder, bias representation model (BRM), bias-aware user modeling (BAUM), and bias-aware click predictor (BACP). Their details are introduced as follows.

4.1 Content Encoder

In this section, we introduce the content encoder module in DebiasRec, which is based on the title encoder in NAML [39]. The content encoder module is used to learn news content representations, which contains three layers. The first layer is an embedding layer, which converts the words within a news title into a sequence of low-dimensional word embeddings. The second layer is a convolutional neural network (CNN), which captures the local contexts of each word to learn their contextual representations. For the i_{th} word in the news title, its contextual word representation learned by this CNN layer is denoted as \mathbf{h}_i^w . The third layer is a word-level attention network, which is used to select important words within the same news title for learning informative news representations. We denote the attention weight of the i_{th} word as α_i^w , which is computed as follows:

$$\begin{aligned} a_i^w &= \mathbf{q}_w^T \tanh(\mathbf{V}_w \times \mathbf{h}_i^w + \mathbf{v}_w), \\ \alpha_i^w &= \frac{\exp(a_i^w)}{\sum_{j=1}^L \exp(a_j^w)}, \end{aligned} \quad (1)$$

where \mathbf{q}_w is a learnable query vector in the word-level attention network, \mathbf{V}_w and \mathbf{v}_w are projection parameters, and L is the length of the news title. The final representation \mathbf{c} of a news title is the

Figure 3: The framework of our *DebiasRec* method.

weighted summation of its contextual word representations, i.e., $\mathbf{c} = \sum_{i=1}^L \alpha_i^w \times \mathbf{h}_i^w$.

4.2 Bias Representation Model

Next, we introduce the bias representation model (BRM) in our approach, which aims to learn accurate bias representations from multiple kinds of bias information and meanwhile consider their interactions. As shown in Figure 2, the position and the size have strong impacts on user click behaviors. Therefore, in our bias representation model, we model the impacts of both biases. For the position p of each news, we use a position embedding layer to convert it into a low-dimensional position embedding \mathbf{e}_p . For the size of news s , we use another size embedding layer to convert it into its size embedding \mathbf{e}_s . A simple way to generate bias representations is to utilize a linear perceptron layer to transform the concatenate of \mathbf{e}_p and \mathbf{e}_s into a unified bias representation \mathbf{b} . The bias representation \mathbf{b} can be computed as follows:

$$\mathbf{b} = \mathbf{W}_b \times [\mathbf{e}_p || \mathbf{e}_s] + \mathbf{b}_b, \quad (2)$$

where $||$ is concatenate operation, \mathbf{W}_b and \mathbf{b}_b are projection parameters. However, the position bias and the size bias are not independent of each other (verified by the Chi-square test result

in Section 5.4). For example, large news is likely to be displayed at special and conspicuous positions, i.e., in the middle or top of the screen. Researchers have found that the add operation of the perceptron layer used in Eq. 2 is weak in capturing the interactions between variables [22, 26, 31]. The bias effects may not be effectively modeled if their interactions are not fully considered. Inspired by Qu et al. [26], we use the element-wise product to help capture the dependencies between different biases. In our method, the unified bias representation \mathbf{b} is computed as follows:

$$\begin{aligned} \mathbf{e}_i &= \mathbf{e}_p \odot \mathbf{e}_s, \\ \mathbf{b} &= \mathbf{W}_b \times [\mathbf{e}_p || \mathbf{e}_s || \mathbf{e}_i] + \mathbf{b}_b, \end{aligned} \quad (3)$$

where \odot stands for the element-wise product operation. In this way, the bias representation encodes not only the information of each kind of bias, but also the interactions between different kinds of biases, which may be helpful for effective debiasing.

4.3 Bias-aware User Modeling

The bias-aware user modeling (BAUM) module is used to learn debiased user interest representations from click behaviors by distilling the real user preferences from them with the help of bias information. Users' click behaviors do not necessarily indicate user

interests because they are heavily influenced by many bias information. Users may click news due to the attraction of news displayed at top positions or in large size rather than their personal interests. For example, in Figure 2, a user may be “attracted” to click the first news, which is the most conspicuous one in the webpage. It is difficult to identify whether the user is actually interested in the content of this news. However, if a user clicks the third news “Tommy Li says he will leave the US if trump wins” which is displayed with small size and low position, we can infer that the user is probably interested in this news. Thus, selecting clicked news that can better indicate user interests is important for reducing the bias information in user modeling and calibrating the interest model.

Motivated by these observations, we propose a bias-aware attention network to debias the click behaviors and learn debiased user interest representations. For each clicked news n_i , we first compute its content representation \mathbf{c}_i using the content encoder and compute its bias representation \mathbf{b}_i according to its position p_i and size s_i using the bias representation model. The bias-aware attention network takes both the content and bias representations of clicked news as input to calculate bias attention scores and content attention scores. The bias attention mechanism aims to estimate the expected influence of bias factors on click probabilities and reduce bias effects on user modeling. We denote the bias attention score of the i -th news as a_i^b , which is computed as follows:

$$a_i^b = \mathbf{q}_b^T \tanh(\mathbf{V}_b \times \mathbf{b}_i + \mathbf{v}_b), \quad (4)$$

where \mathbf{q}_b is a learnable query vector in the bias attention network, and \mathbf{V}_b and \mathbf{v}_b are projection parameters.

The content attention mechanism aims to evaluate the informativeness of the content of clicked news for user interest modeling. According to previous studies, different news usually contributes differently toward modeling user interests [39, 41, 46]. For example, the title ‘Pulse Extra’ is much less informative for inferring user interest than another title ‘Spectacular photos from space’. Therefore, we use a content attention network to select informative click behaviors. It computes the content attention scores of clicked news based on the their content representations. We denote the content attention score of the i -th clicked news as a_i^c , which is computed as follows:

$$a_i^c = \mathbf{q}_c^T \tanh(\mathbf{V}_c \times \mathbf{c}_i + \mathbf{v}_c), \quad (5)$$

where \mathbf{q}_c is the query vector of content attention network, and \mathbf{V}_c and \mathbf{v}_c are parameters.

Then, the content attention scores and bias attention scores are fused into unified attention weights that indicate the relative importance of different clicked behaviors. We denote the unified attention weight of the i -th click behavior as α_i^n , the user representation as \mathbf{u} , which are calculated as follows:

$$\alpha_i^n = \frac{\exp(a_i^c + a_i^b)}{\sum_{j=1}^M \exp(a_j^c + a_j^b)}, \quad (6)$$

The final user representation \mathbf{u} is the weighted summation of the element product between the content and bias representations of

clicked news, which is formulated as:

$$\mathbf{u} = \sum_{i=1}^M \alpha_i^n \times (\mathbf{b}_i \odot \mathbf{c}_i), \quad (7)$$

where \odot represents the element-wise product operation.

4.4 Bias-aware Click Prediction

The bias-aware click prediction (BACP) module aims to help learn a debiased news recommendation model. Besides the user interests, some interest-independent bias factors also strongly influence user click behaviors, such as the displayed position and size of news. As shown in Figure 2, the click-through rate decreases rapidly with the positions and increases with the sizes. Several eye tracking studies [9, 10, 24] also show the same conclusion. Therefore, click labels contain much bias information, and the directly trained with the biased click data will inherit the unwanted biases, which is harmful for accurate user interest matching. To solve this problem, we propose a bias-aware click predictor to model the bias effects on click labels and compute bias-free user preference scores. In the bias-aware click predictor, we decompose the overall click score s_c into a summation of a preference score s_p that indicates user interests on a candidate news and a bias score s_b that indicates the influence of bias on click probability. The preference score s_p is computed by the dot product between the user representation and candidate news representation, and the bias scores s_b is predicted from the bias representation \mathbf{b} of candidate news, which are formulated as follows:

$$\begin{aligned} s_b &= \mathbf{W}_{pb} \times \mathbf{b} + \mathbf{b}_{pb}, \\ s_p &= \mathbf{u}^T \mathbf{c}, \\ s_c &= s_b + s_p, \end{aligned} \quad (8)$$

where \mathbf{W}_{pb} and \mathbf{b}_{pb} are projection parameters. Note that only the preference score s_p is used to rank candidate news in the test stage, i.e., only historical clicked behaviors and the content information of candidate news are used for test. In this way, DebiasRec can generate debiased click prediction scores for recommendation.

4.5 Model Training

Following many previous methods [14, 41], we apply negative sampling strategy to build training samples. For every positive sample (clicked news), we randomly sample K negative samples (non-clicked news) from the same impression. We denote the prediction click scores, preference scores and bias scores of these $K+1$ sample as $\{(s_c^i, s_b^i, s_p^i) | i = 1, 2, \dots, K+1\}$, and denote the click labels of the samples as $\{y^1, y^2, \dots, y^{K+1}\}$ (1 for click and 0 for non-click). In the training process, we first use the softmax function to normalize each click score into a click probability, which is computed as:

$$p^i = \frac{\exp(s_c^i)}{\sum_{k=1}^{K+1} \exp(s_c^k)}, \quad (9)$$

To maximize the click probability of the positive sample, we use the cross-entropy loss as the loss function, which is formulated as:

$$\mathcal{L} = - \sum_{i=1}^{K+1} y^i \times \log(p^i). \quad (10)$$

Table 1: Results of different methods. *Improvements over other baselines are significant ($p < 0.05$ in t-test).

Methods	News				Feeds			
	AUC	MRR	nDCG@5	nDCG@10	AUC	MRR	nDCG@5	nDCG@10
LibFM	58.26±0.06	19.04±0.04	18.91±0.04	23.48±0.04	60.82±0.02	28.57±0.03	30.20±0.04	38.18±0.02
DSSM	60.95±0.15	20.25±0.14	20.35±0.12	25.06±0.12	62.30±0.33	29.71±0.35	31.59±0.41	39.56±0.36
DeepFM	58.32±0.35	18.54±0.26	18.40±0.25	23.09±0.25	62.56±0.16	30.06±0.11	32.00±0.12	39.87±0.14
Wide&Deep	60.05±0.17	19.59±0.21	19.61±0.24	24.33±0.23	62.82±0.05	30.07±0.08	32.05±0.11	39.97±0.09
DFM	60.22±0.08	19.78±0.15	19.82±0.14	24.50±0.12	62.71±0.16	30.17±0.11	32.15±0.15	39.99±0.11
DKN	61.14±0.15	19.99±0.34	20.02±0.29	24.87±0.28	63.37±0.15	30.44±0.16	32.53±0.20	40.42±0.17
NAML	62.65±0.20	21.55±0.13	21.94±0.15	26.76±0.17	64.10±0.12	31.20±0.07	33.46±0.06	41.27±0.06
NRMS	62.76±0.18	21.56±0.16	21.98±0.17	26.84±0.19	63.91±0.36	31.05±0.30	33.24±0.36	41.09±0.34
REM	62.88±0.21	21.66±0.14	22.24±0.17	26.89±0.22	64.34±0.15	31.37±0.17	33.61±0.13	41.43±0.16
PAL	62.96±0.36	21.74±0.19	22.13±0.24	26.97±0.24	64.45±0.09	31.45±0.06	33.77±0.06	41.55±0.07
DebiasRec*	63.70±0.15	22.33±0.17	22.81±0.17	27.67±0.15	65.34±0.07	32.31±0.10	34.83±0.12	42.57±0.10

Table 2: Detailed statistics of News and Feeds datasets.

	# News	# Users	# Impressions	# Click	# Sizes
News	97,646	310,469	58,000,000	917,839	4
Feeds	643,177	10,000	320,925	970,846	4

5 EXPERIMENTS

In this section, we demonstrate the effectiveness of our approach by answering the following four research questions:

- **RQ1:** How does DebiasRec perform compared with baseline methods?
- **RQ2:** Are BAUM and BACP both useful?
- **RQ3:** Can BRM model different biases and their interactions for accurate news bias modeling?
- **RQ4:** Can DebiasRec provide reasonable recommendation results?

5.1 Datasets and Experimental Settings

Since there is no off-the-shelf news recommendation dataset with position bias and presentation bias information, we build two real-world datasets from the user logs of two news platforms, named **News and Feeds**. *News* is collected on Microsoft News website from October 13th, 2019 to November 13th, 2019. *Feeds* is collected on Microsoft News App from August 1st, 2020 to September 1st, 2020. The detailed dataset statistics are summarized in Table 2. For both datasets, the impressions in the first three weeks are used for model training, and the rest impressions in the last week are used for test. Additionally, we randomly sample 20% impressions in the training set for validation. Note that the bias features of candidate news are not used for click prediction in the test stage, while the bias features of user historical clicked news are used in both training and test stages to debias user click behaviors. Following many previous news recommendation works [39, 43], we use AUC, MRR, nDCG@5 and nDCG@10 as the evaluation metrics.

In our experiments, we apply pretrained Glove embedding [25] to initialize the word embedding matrix. The dimension of word embedding vectors is 300. The dimension of position and size embedding vectors is 200. The number of filters in CNN layer is 400,

and the window size is 3. We apply dropout to each layer to mitigate overfitting. The dropout rate is 0.2. Adam [16] is used for model optimization. The batch size is set to 32. The number of negative samples associated with each positive sample is 4. The max position is set to 400 in both datasets. All hyper-parameters are selected according to results on the validation set. We repeat each experiment 5 times independently, and we report the average results with standard deviations.

5.2 Performance Evaluation (RQ1)

We evaluate the performance of DebiasRec by comparing it with two groups of baselines. The first group consists of several baseline methods for news recommendation, including:

- **LibFM** [27] a factorization machine based recommendation method. In this paper, we use TF-IDF features of candidate news and TF-IDF features of user clicked news as inputs.
- **DSSM** [14] a deep structured semantic model. In this work, we regard candidate news as documents and user clicked news as queries.
- **DeepFM** [11] a factorization-machine based deep network, which contains a FM channel and deep channel.
- **DFM** [18] a multi-channel deep fusion model for personalized news recommendation. In this work, TF-IDF features of user clicked news and candidate news are treated as inputs.
- **Wide&Deep** [5] a deep learning based recommendation method that combines a wide channel and a deep channel. TF-IDF features of candidate news and user clicked news are used as inputs.
- **DKN** [34] deep knowledge-aware network for news recommendation.
- **NAML** [39] a neural news recommendation methods with attentive multi-view learning.
- **NRMS** [41] a neural news recommendation method that uses self-attention for news and user modeling.

The second group consists of several baseline methods for click data debiasing, including:

- **REM** [37]: regression-based EM algorithm for unbiased learning to rank. We use NAML as the predictor model in REM.

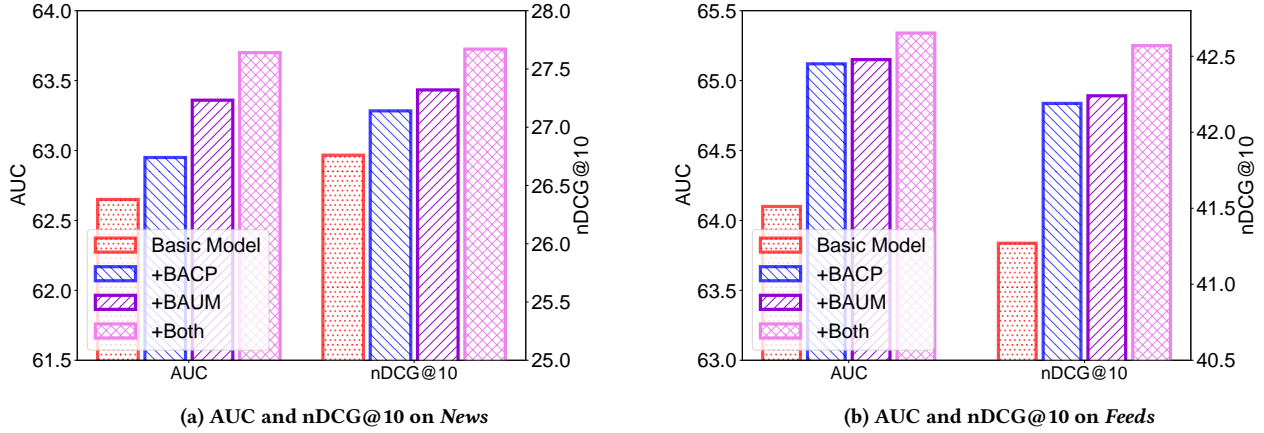


Figure 4: Effectiveness of BAUM and BACP.

- **PAL** [12]: position-bias aware learning framework for CTR prediction. In training stage, a position module is used for modeling position bias. In testing process, the position module is removed.

In all baseline methods, news are represented by their titles for fair comparison. The results are shown in Table 1, from which we have two major findings. First, compared with the methods (e.g., NAML and NRMS) that do not consider the bias information in news clicks, bias-aware recommendation methods (e.g., PAL and DebiasRec) perform better. This is because the bias information in click behaviors will affect user interest modeling and model training, which is usually not beneficial for targeting user interests. Second, our DebiasRec method consistently outperforms all baseline methods on both dataset, and the results of t-test show the improvements are significant. This is because our DebiasRec can alleviate the bias effect in both user modeling and click prediction, which can help learn debiased user interest representations and click prediction model. In addition, our DebiasRec can model multiple kinds of bias (i.e. position bias and size bias in our experiments) and further consider the interactions between them. Therefore, DebiasRec is able to better eliminate bias effect in user interest modeling and click prediction.

5.3 Effectiveness of BAUM and BACP (RQ2)

In this section, we further conduct several experiments to validate the effectiveness of several core components in DebiasRec, i.e., bias-aware user modeling (BAUM) and bias-aware click prediction (BACP). We compare the recommendation performance of DebiasRec and its variants with one of these components removed. The results on *News* and *Feeds* are shown in Figure 4. We have several observations from the results. First, incorporating the bias-aware click prediction module can improve model performance on both datasets. This is because in real-world recommendation scenarios, interest-independent biases such as position bias and size bias have strong impacts on user click behaviors. Thus, the click labels cannot fully reflect user preference. Our DebiasRec decomposes the click score into a bias score to model the bias effect on news click and a preference score to indicate bias-independent user interests,

and we only use the preference score for news ranking. Thus, the bias effect on click behaviors on model training can be effectively reduced. In addition, the bias-aware user modeling can also consistently improve the performance. This is because the historical user click behaviors for user interest model are also biased. Our DebiasRec can further debias the historical click behaviors in user interest modeling to learn debiased user representations, which can benefit the subsequent user interest matching. Besides, incorporating both BACP and BAUM modules can further improve the recommendation performance, which validate their effectiveness.

5.4 Effectiveness of BRM (RQ3)

In this section, we first demonstrate the dependency of position bias and size bias based on *News* dataset. Then we investigate how modeling the interaction between different biases contributes to our method.

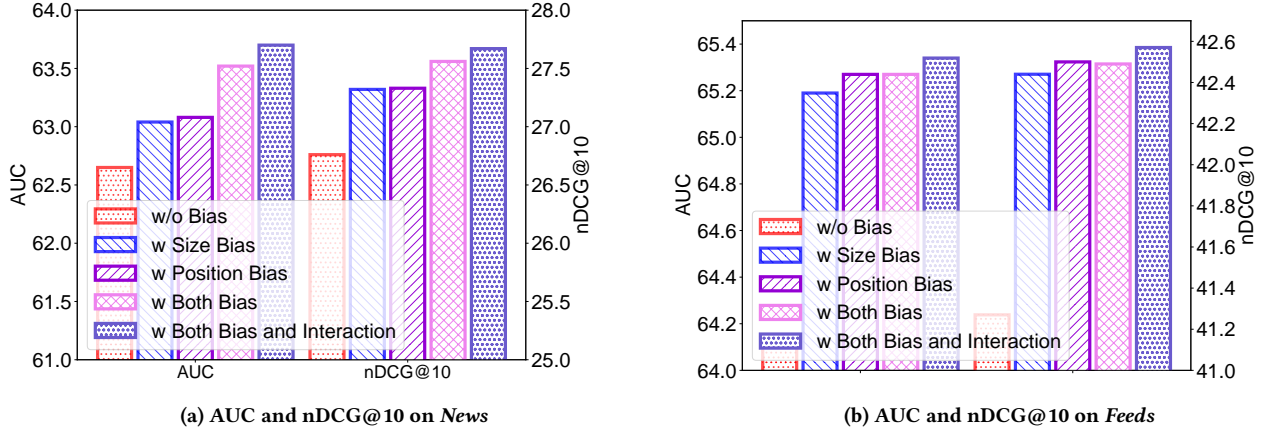
Dependency between position and size. In most scenarios, different kinds of biases are not independent to each other and they usually have joint effects on user behaviors. To show this phenomenon, we test the dependency of position bias and size bias on the *News* dataset using the Chi-square table. Table 3 shows the distribution of news displayed in different positions and sizes. The p-value of Chi-square test is less than 0.01, which indicates the news positions and news sizes are not independent of each other on *News*.⁵

We then conduct an ablation study to verify the effectiveness of modeling multiple kinds of biases (i.e. position bias and size bias) and the interactions between them. We compare the recommendation performance of DebiasRec on *News* and *Feeds* and its variants with one of these biases or their interactions removed. We have several observations from Figure 5. First, when the position bias is added, further incorporating the size bias can effectively improve the model performance on the *News* dataset, while the improvement is marginal on *Feeds*. After carefully studying, we find the user interface on the platform of *Feeds* is static, but the user interface in *News* is dynamic. For example, the news in first position and second position in *Feeds* only displayed with a fixed type “NewsDigest”.

⁵We have the same finding on the *Feeds* dataset.

Table 3: The distribution of news in different positions and sizes.

Size	Position									
	1	2	3	4	5	6	7	8	9	10
mini	10,989	12,659	14,399	72,671	46,262	37,664	17,379	63,947	24,4801	254,513
small	472,250	283,015	407,429	298,987	279,691	29,6784	218,015	93,482	13,3607	115,277
medium	10,893	10,454	38,405	73,495	51,084	24,293	18,094	11,873	18,650	25,685
large	16,876	5,315	12,166	26,533	34,471	21,106	22,500	10,495	10,080	8,620

**Figure 5: Effectiveness of modeling different kinds of biases and their interactions.**

Content Information	Preference Score	Bia Information		Bias Score
		Size	Position	
Christopher Dennis Dies: 'Hollywood Superman' And Frequent 'Jimmy Kimmel Live!' Guest Was 52	0.64	2	45	0.23
How to Avoid Airbnb Scams and Find Legit Vacation Rentals Online	0.37	2	7	0.49
Dubrovnik Has So Many Tourists It May Ban New Restaurants for Five Years	0.25	2	10	0.39

Figure 6: Visualization of preference scores and click scores in BACP.

The tenth position in *Feeds* is assigned the displayed type “Video-Card”. In *Feeds*, given a position p , the size is known. Therefore, the results in 5b match the real-world scenarios. Second, either adding position bias or size bias outperforms the method with no bias in both datasets. It makes sense since user click behaviors are affected by both the position and the size bias in real-world recommendation scenario. Alleviating the bias effect of either of them can help better model user interest. Third, considering both kinds of biases can further improve the performance in *News*. This is because user click behaviors are affected by several kinds of biases in real-world recommendation scenario. Additionally, the user interface in *News* is dynamic, so considering two kinds of the bias information can help model the bias effect more precisely than considering single bias. Forth, the bias interaction module helps in *News*. This is because the position and size are not totally independent to each other. Compared with perceptron layer, the element-wise product in bias representation model (BRM) has better performance in exploring the interaction between different features.

5.5 Case Study (RQ4)

In this section, we first provide a case study on what the bias scores and preference scores will the bias-aware click predictor assign to different candidate news. To explore how the bias-aware predictor works, we randomly select a user who prefers clicking entertainment news, such as “Keanu Reeves holds hands with Alexandra Grant as the Internet swoons”, “Jamie Lee Curtis Opens Up About Being 20 Years Sober, Going Public With Her Addiction” and “Caitlyn Jenner, Khloe Kardashian’s relationship remains strained as Caitlyn turns 70”. We randomly select three candidate news and compute their preference scores and bias scores in the bias-aware click predictor. The first news is entertainment news about an actor, and is displayed at a lower position by the user. The second and the third news are about vacation and travel which are irrelevant to entertainment, and they are displayed at higher positions than the first news. The first and the second news are clicked by the user, while the third is not clicked. The prediction results are shown in Figure 6. We can observe that the bias-aware click predictor assign the highest preference score to the first news. This is because our

User	Content Information	Content Attention	Bias Information		Bias Attention
			Size	Position	
1	Rise and Grind: A Hot Dog Success Story	1.81	2	23	0.62
	Far-reaching snowstorm may take shape over US	-1.05	3	1	0.57
2	Trailer - Hoax	-0.49	2	30	0.69
	'Angel' car saves Arizona family after smashing into suspected drunk driver running red light, video shows	1.07	3	5	0.58

Figure 7: Visualization of attention weights in BAUM.

DebiasRec can capture user interest, and the first news is of his/her interested category. We can also find that the first news is assigned the lowest bias score among all three news. This is because our DebiasRec can assign bias score according to the bias effect. The final click score of the first and second news is higher than the third news, which show our DebiasRec can predict click behaviors more precisely after considering the bias effect.

Then we conduct a case study to show how the bias-aware user encoder selects user clicked news based on both news contents and bias factors. We random sample two users and two news in their historical clicked behaviors. The results are shown in Figure 7. We have several observations from the cases. First, the bias-aware user encoder can reasonably debias the user click behaviors from the bias factors. The first news of the first and second user are shown in small sizes and low positions, which indicates the users are interested in the news contents. Therefore, they are assigned high bias attention scores. The second news of both users are shown in larger sizes and higher positions, which are assigned lower bias attention scores. Second, the bias-aware user encoder can select more informative news to model user interest according to news content information. For the clicked news of the first user, the first news is about food and drink which indicates the user may be interested in making food, and the second news is a regular weather forecast news which is less informative. Therefore, our bias-aware user encoder computes a low content attention score for the first news and a high score for the second one. For the clicked news of the second user, even though the first news is a movie name, it is hard for our model to guess user interests from two words. Therefore, it is assigned low content attention score. This observation also shows our content encoder cannot understand entity information, which inspires us to perform knowledge-aware content encoder in our future work.

6 CONCLUSION

In this paper, we propose a bias-aware personalized news recommendation method named DebiasRec, which can effectively reduce the bias effect on user interest inference and model training. The core of our approach is a bias representation model (BRM), a bias-aware user modeling model (BAUM), and a bias-aware click prediction model (BACP). BRM is used to learn bias representation from different kinds of news biases and meanwhile model their interactions to capture their joint effect on click behaviors. BAUM aims to model debiased user interests from clicked news articles by using their bias information to calibrate the user interest model. BACP is used to train an debiased news recommendation model from the biased click behaviors by decomposing the click score

into a preference score that indicates user interest in news content and a bias score inferred from its various bias features. Extensive experiments on two real-world datasets show that our method can effectively improve the performance of news recommendation.

REFERENCES

- [1] Mingxiao An, Fangzhao Wu, Chuhan Wu, Kun Zhang, Zheng Liu, and Xing Xie. 2019. Neural news recommendation with long-and short-term user representations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 336–345.
- [2] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research* 3, Jan (2003), 993–1022.
- [3] Olivier Chapelle and Ya Zhang. 2009. A dynamic bayesian network click model for web search ranking. In *Proceedings of the 18th international conference on World wide web*. 1–10.
- [4] Danqi Chen, Weizhu Chen, Haixun Wang, Zheng Chen, and Qiang Yang. 2012. Beyond ten blue links: enabling user click modeling in federated web search. In *Proceedings of the fifth ACM international conference on Web search and data mining*. 463–472.
- [5] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishikesh Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Isipir, et al. 2016. Wide & deep learning for recommender systems. In *DLRS*. ACM, 7–10.
- [6] Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *Proceedings of the 2008 international conference on web search and data mining*. 87–94.
- [7] Abhinandan S Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. 2007. Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th international conference on World Wide Web*. 271–280.
- [8] Georges E Dupret and Benjamin Piwowarski. 2008. A user browsing model to predict search engine click data from past observations. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. 331–338.
- [9] Laura Granka, Matthew Feusner, and Lori Lorigo. [n.d.]. Eyetracking in online search. ([n. d.]).
- [10] Zhiwei Guan and Edward Cutrell. 2007. An eye tracking study of the effect of target rank on web search. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 417–420.
- [11] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. In *AAAI*. AAAI Press, 1725–1731.
- [12] Huifeng Guo, Jinkai Yu, Qing Liu, Ruiming Tang, and Yuzhou Zhang. 2019. PAL: a position-bias aware learning framework for CTR prediction in live recommender systems. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 452–456.
- [13] Linmei Hu, Siyong Xu, Chen Li, Cheng Yang, Chuan Shi, Nan Duan, Xing Xie, and Ming Zhou. 2020. Graph Neural News Recommendation with Unsupervised Preference Disentanglement. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 4255–4264.
- [14] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *CIKM*. ACM, 2333–2338.
- [15] Thorsten Joachims, Adithy Swaminathan, and Tobias Schnabel. 2017. Unbiased learning-to-rank with biased feedback. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. 781–789.
- [16] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [17] Lei Li, Li Zheng, Fan Yang, and Tao Li. 2014. Modeling and broadening temporal user interest in personalized news recommendation. *Expert Systems with Applications* 41, 7 (2014), 3168–3177.

- [18] Jianxun Lian, Fuzheng Zhang, Xing Xie, and Guangzhong Sun. 2018. Towards better representation learning for personalized news recommendation: a multi-channel deep fusion approach. In *IJCAI*. 3805–3811.
- [19] Xiaoliang Ling, Weiwei Deng, Chen Gu, Hucheng Zhou, Cui Li, and Feng Sun. 2017. Model ensemble for click prediction in bing search ads. In *Proceedings of the 26th International Conference on World Wide Web Companion*. 689–698.
- [20] Jiahui Liu, Peter Dolan, and Elin Rønby Pedersen. 2010. Personalized news recommendation based on click behavior. In *Proceedings of the 15th international conference on Intelligent user interfaces*. 31–40.
- [21] Rui Liu, Huilin Peng, Yong Chen, and Dell Zhang. 2020. HyperNews: Simultaneous News Recommendation and Active-Time Prediction via a Double-Task Deep Neural Network. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence and the 17th Pacific Rim International Conference on Artificial Intelligence (IJCAI-PRICAI)*, Yokohama, Japan.
- [22] Aditya Krishna Menon, Krishna-Prasad Chitrapura, Sachin Garg, Deepak Agarwal, and Nagaraj Kota. 2011. Response prediction using collaborative filtering with hierarchies and side-information. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. 141–149.
- [23] Shumpei Okura, Yukihiko Tagami, Shingo Ono, and Akira Tajima. 2017. Embedding-based news recommendation for millions of users. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1933–1942.
- [24] Bing Pan, Helene A Hembrooke, Geri K Gay, Laura A Granka, Matthew K Feusner, and Jill K Newman. 2004. The determinants of web page viewing behavior: an eye-tracking study. In *Proceedings of the 2004 symposium on Eye tracking research & applications*. 147–154.
- [25] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.
- [26] Yanru Qu, Han Cai, Kan Ren, Weinan Zhang, Yong Yu, Ying Wen, and Jun Wang. 2016. Product-based neural networks for user response prediction. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 1149–1154.
- [27] Steffen Rendle. 2012. Factorization machines with libfm. *TIST* 3, 3 (2012), 57.
- [28] Matthew Richardson, Ewa Dominowska, and Robert Ragno. 2007. Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*. 521–530.
- [29] Chengyao Shen and Qi Zhao. 2014. Webpage Saliency. In *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (Eds.). Springer International Publishing, Cham, 33–46.
- [30] Jeong-Woo Son, A-Yeong Kim, and Seong-Bae Park. 2013. A location-based news article recommendation with explicit localized semantic analysis. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. 293–302.
- [31] Anh-Phuong Ta. 2015. Factorization machines with follow-the-regularized-leader for CTR prediction in display advertising. In *2015 IEEE International Conference on Big Data (Big Data)*. IEEE, 2889–2891.
- [32] Chao Wang, Yiqun Liu, Min Zhang, Shaoping Ma, Meihong Zheng, Jing Qian, and Kuo Zhang. 2013. Incorporating vertical results into search click models. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. 503–512.
- [33] Heyuan Wang, Fangzhao Wu, Zheng Liu, and Xing Xie. 2020. Fine-grained Interest Matching for Neural News Recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 836–845.
- [34] Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2018. DKN: Deep knowledge-aware network for news recommendation. In *Proceedings of the 2018 world wide web conference*. 1835–1844.
- [35] Wenjie Wang, Fuli Feng, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2020. "Click" Is Not Equal to "Like": Counterfactual Recommendation for Mitigating Clickbait Issue. *arXiv preprint arXiv:2009.09945* (2020).
- [36] Xuanhui Wang, Michael Bendersky, Donald Metzler, and Marc Najork. 2016. Learning to rank with selection bias in personal search. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*. 115–124.
- [37] Xuanhui Wang, Nadav Golbandi, Michael Bendersky, Donald Metzler, and Marc Najork. 2018. Position bias estimation for unbiased learning to rank in personal search. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. 610–618.
- [38] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 950–958.
- [39] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. Neural news recommendation with attentive multi-view learning. *arXiv preprint arXiv:1907.05576* (2019).
- [40] Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019. NPA: neural news recommendation with personalized attention. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2576–2584.
- [41] Chuhan Wu, Fangzhao Wu, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. 2019. Neural News Recommendation with Multi-Head Self-Attention. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 6389–6394. <https://doi.org/10.18653/v1/D19-1671>
- [42] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. [n.d.]. User Modeling with Click Preference and Reading Satisfaction for News Recommendation. ([n. d.]).
- [43] Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. 2020. Mind: A large-scale dataset for news recommendation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 3597–3606.
- [44] Liwei Wu, Shuqing Li, Cho-Jui Hsieh, and James Sharpnack. 2020. SSE-PT: Sequential recommendation via personalized transformer. In *Fourteenth ACM Conference on Recommender Systems*. 328–337.
- [45] Bo Zhao, Zhuoming Xu, Yan Tang, Jian Li, Bei Liu, and Haimei Tian. 2020. Effective Knowledge-Aware Recommendation via Graph Convolutional Networks. In *International Conference on Web Information Systems and Applications*. Springer, 96–107.
- [46] Qiannan Zhu, Xiaofei Zhou, Zeliang Song, Jianlong Tan, and Li Guo. 2019. Dan: Deep attention neural network for news recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 5973–5980.