

DebiasGAN: Eliminating Position Bias in News Recommendation with Adversarial Learning

Chuhan Wu¹ Fangzhao Wu² Xiangnan He³ Yongfeng Huang^{1,4}

¹Department of Electronic Engineering, Tsinghua University, Beijing 100084, China

²Microsoft Research Asia, Beijing 100080, China

³University of Science and Technology of China, Hefei 230027, China

⁴Zhongguancun Laboratory, Beijing 100094, China

{wuchuhan15, wufangzhao, xiangnanhe}@gmail.com,
yfh Huang@tsinghua.edu.cn

Abstract

Click behaviors are widely used for learning news recommendation models, but they are heavily affected by the biases brought by the news display positions. It is important to remove position biases to train unbiased recommendation model and capture unbiased user interest. In this paper, we propose a news recommendation method named *DebiasGAN* that can effectively alleviate position biases via adversarial learning. The core idea is modeling the personalized effect of position bias on click behaviors in a candidate-aware way, and learning debiased candidate-aware user embeddings from which the position information cannot be discriminated. More specifically, we use a bias-aware click model to capture the effect of position bias on click behaviors, and use a bias-invariant click model with random candidate positions to estimate the ideally unbiased click scores. We apply adversarial learning to the embeddings learned by the two models to help the bias-invariant click model capture debiased user interest. Experimental results on two real-world datasets show that *DebiasGAN* effectively improves news recommendation by eliminating position biases.

1 Introduction

Accurate news recommendation is critical for improving users' online news reading experience (Wu et al., 2020b). Existing news recommendation methods mainly use users' news click behaviors for interest inference and model training (Wang et al., 2018a; Wu et al., 2019a; Ge et al., 2020; Hu et al., 2020; Wu et al., 2020a, 2021a; Qi et al., 2021b,a). However, news click behaviors are heavily influenced by position biases, i.e., the positions of news when displayed on a webpage (Chen et al., 2020; Bhadani, 2021; Xun et al., 2021). Since top ranked news are more likely to be clicked than those displayed inconspicuously (Baeza-Yates, 2018), models directly learned on click data may be inaccurate in targeting user interest (Yi et al., 2021).

Most existing studies on eliminating position biases in recommendation follow two popular ways (Sato et al., 2020; Liu et al., 2020; Chen et al., 2021; Wu et al., 2021b). The first one is propensity weight, which is a canonical way by weighting the training data based on the influence of position bias (Agarwal et al., 2019), which can be estimated by randomly ranked data (Joachims et al., 2017) or regression-based models (Wang et al., 2018b). The second one is click model, which aims to simulate the generative process of click behaviors (Craswell et al., 2008). It is combined with deep recommendation models by taking positional information as the inputs to disentangle position bias and position-independent user interest (Guo et al., 2019; Huang et al., 2021). However, existing methods usually model position bias in a non-personalized way without considering user characteristics. In fact, the same position may generate different impacts on different users due to their diverse click preferences. For example, some users prefer to skip the top-displayed content (Benway, 1999). Thus, personalized bias modeling can help better model and eliminate the effects of position bias.

In this paper, we propose a news recommendation method named *DebiasGAN*¹, which effectively eliminates position biases in news recommendation via adversarial learning. The core idea of our method is two-fold, i.e., using candidate-aware click models to capture the personalized influence of position biases, and using adversarial learning to enforce the candidate-aware user representations learned on real positions to be indistinguishable from those learned on random positions. More specifically, we use a bias-aware click model to capture position biases and a bias-invariant click model with randomized positions of candidate news to estimate the ideally unbiased click scores. Both models consider the interactions between positions

¹Code is available at <https://github.com/wuch15/DebiasGAN>.

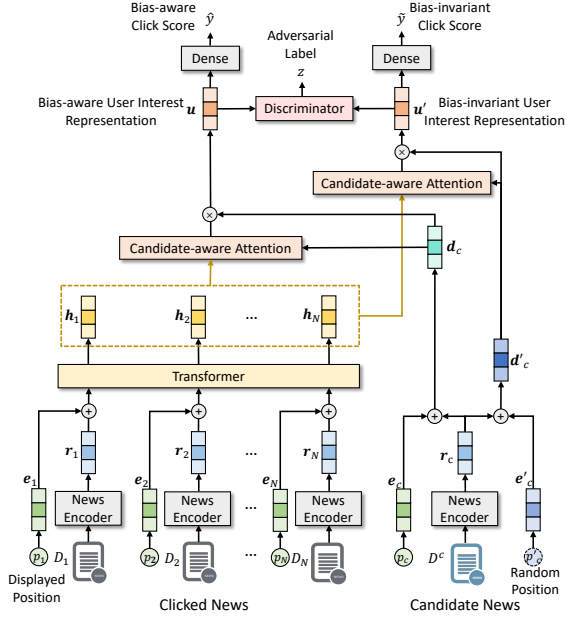


Figure 1: Model architecture of *DeBiasGAN*.

of candidate news and clicked news to achieve personalized bias modeling. Since the bias information is coupled with candidate-aware user interest in the personalized click models, we propose to apply adversarial learning to the hidden representations encoded by the two click models to encourage the bias-invariant click model to capture debiased user interest in candidate news. Experiments on two real-world datasets show that *DeBiasGAN* can effectively reduce position biases to improve the accuracy of news recommendation, especially on the unbiased dataset generated by random ranking.

2 DeBiasGAN

The architecture of *DeBiasGAN* is shown in Fig. 1. It contains a bias-aware click model that captures the position bias effect on click behaviors and a bias-invariant click model to model bias-invariant user interest on candidate news. Both models can consider the interactions between user behaviors and candidate news to learn candidate-aware user representations. Their details are introduced below.

2.1 Bias-aware Click Model

Denote the N historical clicked news of a user as $[D_1, D_2, \dots, D_N]$ and the candidate news as D^c . We use a news encoder to learn semantic representations of news from their texts. Motivated by (Wu et al., 2019c), we use the Transformer (Vaswani et al., 2017) model as the news encoder. We denote the hidden representation of the clicked news D_i

and candidate news D^c as r_i and r_c , respectively. To model the impact of position biases on user interest modeling, we incorporate the embedding of the displayed positions of clicked news. We denote the position of the clicked news D_i as p_i (starts from 1). To reduce the sparsity of positions, we quantize each position p_i by $\hat{p}_i = \lceil \sqrt{p_i - 1} \rceil$. We convert the quantized position \hat{p}_i into its embedding e_i , and add it to the semantic news embedding r_i to obtain bias-aware news representations. We also add position embedding of the candidate news (denoted as e_c) to r_c to obtain a bias-aware candidate news representation d_c . The bias-aware representations of clicked news are further process by a behavior Transformer, which captures the relations between click behaviors to help better model interests. We denote its output as $\mathbf{H} = [h_1, h_2, \dots, h_N]$.

Since the same position may generate different impacts on different users, we incorporate user preferences into the modeling of the effects of position bias on click behaviors. More specifically, we use a candidate-aware attention network to select important click behaviors to accurately model user interests on the candidate news displayed at a given position. We denote the bias-aware user interest representation with respect to the candidate news as \mathbf{u} , which is formulated as follows:

$$\begin{aligned} \mathbf{a} &= \text{softmax}(\mathbf{d}_c \tanh(\mathbf{W}_c \mathbf{H} + \mathbf{w}_c)), \\ \mathbf{u} &= \mathbf{d}_c \odot (\mathbf{H} \mathbf{a}), \end{aligned} \quad (1)$$

where \mathbf{W}_c and \mathbf{w}_c are parameters, \mathbf{a} is the candidate-aware attention weights, and \odot means element-wise product. We further predict a bias-aware click score \hat{y} based on \mathbf{u} , which is formulated as $\hat{y} = \mathbf{w}^\top \mathbf{u}$, where \mathbf{w} is a parameter vector. This score indicates the predicted probability of a user clicking on a candidate news given the position of candidate news and the user’s personal preference on the content and position of this news.

2.2 Bias-invariant Click Model

The bias-invariant click model aims to estimate the ideally unbiased click scores. It shares the news encoder and the behavior Transformer with the bias-aware click model, while it adds the embedding of a random candidate news position (denoted as e'_c) to the candidate news representation r_c to form a bias-randomized representation d'_c .² We use another candidate-aware attention network that uses d'_c as

²In the test phase, we add the embedding of the default position zero.

attention query to learn bias-invariant user interest representation \mathbf{u}' on the candidate news as follows:

$$\begin{aligned} \mathbf{a}' &= \text{softmax}(\mathbf{d}'_c \tanh(\mathbf{W}'_c \mathbf{H} + \mathbf{w}'_c)), \\ \mathbf{u}' &= \mathbf{d}'_c \odot (\mathbf{H} \mathbf{a}'), \end{aligned} \quad (2)$$

where \mathbf{W}'_c and \mathbf{w}'_c are parameters, and \mathbf{a}' is the attention weight vector. Since the candidate news position is randomly generated, the bias-invariant click model is encouraged to model user interests that are independent of displayed positions of news. We predict the bias-invariant click score by $\tilde{y} = \mathbf{w}^\top \mathbf{u}'$, where the parameter \mathbf{w} is shared with the bias-aware click model.

2.3 Debiasing with Adversarial Learning

Since the position information of candidate news is coupled with fine-grained user interest information encoded by different user behaviors, it is difficult to apply existing coarse-grained debiasing methods such as propensity weight and click models to reduce the bias effects. Thus, to eliminate position bias in the bias-invariant click model to help it estimate unbiased click scores, we propose to apply adversarial learning techniques to the bias-aware and bias-invariant user interest representations. We use a discriminator to classify whether \mathbf{u} or \mathbf{u}' is learned by the bias-aware model. The adversarial label is predicted by the discriminator as follows³:

$$z = \sigma(\mathbf{w}_p^T \mathbf{u} - \mathbf{w}_n^T \mathbf{u}'), \quad (3)$$

where \mathbf{w}_p and \mathbf{w}_n are parameters, and σ is the sigmoid function. The adversarial loss function \mathcal{L}_A we used is written as $\mathcal{L}_A = -\log(z)$. By propagating the negative gradients of the adversarial loss, both bias-aware and bias-invariant click models are encouraged to learn similar representations of user interests on candidate news. Thus, the distributions of their predicted click scores are expected to be similar, and thereby the effects of position biases on click prediction can be effectively mitigated. Some careful readers may find that the model can simply learn the same embedding for all positions to minimize the adversarial loss. In fact, the click prediction task based on the bias-aware click scores enforces the position embeddings to be informative. Thus, this collapsed case can be avoided by assigning proper loss weights to balance the intensity of click prediction loss and adversarial loss.

Following (Wu et al., 2019b, 2020b), we construct training samples via negative sampling and

³Similar to the formulation of BPR (Rendle et al., 2012).

we use crossentropy as the loss function for click prediction (Wu et al., 2019c). We denote the click prediction losses of the bias-aware and bias-invariant click models as \mathcal{L}_B and \mathcal{L}_D , respectively. The unified loss \mathcal{L} is formulated as follows:

$$\mathcal{L} = \mathcal{L}_B + \mathcal{L}_D - \alpha \mathcal{L}_A, \quad (4)$$

where α is a hyperparameter that controls the relative intensity of adversarial training for debiasing.

3 Experiments

3.1 Datasets and Experimental Settings

We conducted experiments on two datasets. The first one is collected from a commercial news App (denoted as *NewsApp*). The first dataset contains click logs collected from 09/01/2020 to 10/02/2020. The logs in the last week are used for test and the rest are for training and validation (logs in the last 3 days). The second dataset (denoted as *Uniform*) is collected on a commercial news website. It includes click logs collected from 07/12/2020 to 02/01/2021. The training and validation sets are constructed from biased click logs, while the test set is composed of user click behaviors in the last month on uniformly and randomly ranked news lists. Dataset statistics are shown in Table 1. We show the distributions of the displayed positions of news in Fig. 2 and the click-through rate (CTR) of news displayed at different positions in Fig. 3. We can see that the news are not linearly displayed because the CTRs have some periodic patterns with positions and do not simply decline.

	# News	# Users	# Impressions	# Click	# Pos.
<i>NewsApp</i>	590,485	10,000	351,581	493,266	3,214
<i>Uniform</i>	1,277,315	479,281	529,482	808,365	667

Table 1: Statistics of *NewsApp* and *Uniform* datasets.

Following many prior works (Wu et al., 2019b,c; Wang et al., 2020) we use Glove (Pennington et al., 2014) embeddings in the news encoder. The model optimizer is Adam (Kingma and Ba, 2015). The coefficient α is 0.5. The hyperparameters are tuned on validation sets. We use AUC, MRR, nDCG@5 and nDCG@10 to evaluate model performance. We repeat each experiment 5 times. The average scores with standard deviations are reported.

3.2 Performance Comparison

We compare *DebiasGAN* with many baseline news recommendation methods, including *EBNR* (Okura

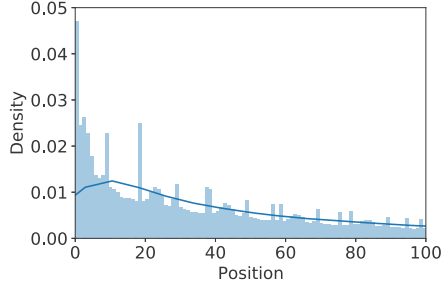


Figure 2: Position distributions in the *NewsApp* dataset.

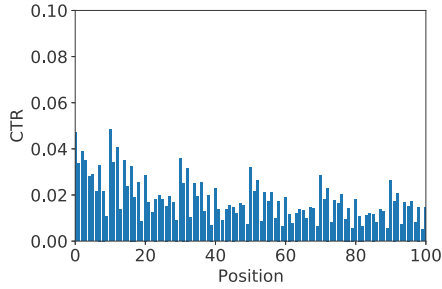


Figure 3: The click-through rate of news displayed at different positions in the *NewsApp* dataset.

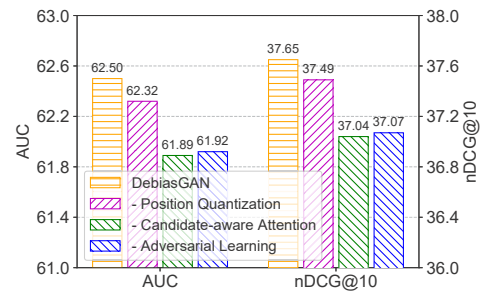
et al., 2017), *DKN* (Wang et al., 2018a), *NAML* (Wu et al., 2019a), *NPA* (Wu et al., 2019b), *LSTUR* (An et al., 2019), and *FIM* (Wang et al., 2020). We also compare several methods for eliminating position biases, including (1) *IPW* (Joachims et al., 2017; Wu et al., 2021b), using inverse propensity weighting in model learning; (2) *Reg-EM* (Wang et al., 2018b), a regression based EM method to estimate the propensity weight; (3) *PAL* (Guo et al., 2019), a position bias-aware learning method for CTR prediction. (4) *DPIN* (Huang et al., 2021), a deep position interaction network for CTR prediction. For fair comparison, in these methods we use the same bias-aware click model with our approach. The results the two datasets are shown in Tables 2 and 3, respectively. We find that debiased methods outperform those directly learned on biased click data. This is because removing position biases can help learn more accurate click prediction model. In addition, our approach outperforms other debiasing methods, and the improvement on the *Uniform* dataset is greater. This is because our approach considers the user preference for position biases to better model and eliminate position biases. Moreover, our approach uses adversarial learning to help model unbiased user interests. Thus, our model yields larger performance gains when the test data is unbiased.

Methods	AUC	MRR	nDCG@5	nDCG@10
EBNR	60.92±0.17	26.12±0.18	27.86±0.19	36.25±0.22
DKN	60.77±0.22	25.91±0.20	27.68±0.19	36.06±0.24
NAML	61.24±0.12	26.10±0.15	28.11±0.14	36.42±0.13
NPA	61.35±0.17	26.24±0.15	28.23±0.18	36.55±0.19
NRMS	61.55±0.11	26.47±0.10	28.47±0.14	36.77±0.13
FIM	61.78±0.16	26.69±0.13	28.70±0.16	36.98±0.16
IPW	61.88±0.13	26.78±0.14	28.81±0.16	37.09±0.15
REM	61.94±0.16	26.83±0.16	28.89±0.15	37.14±0.17
PAL	62.02±0.15	26.89±0.12	28.95±0.14	37.21±0.15
DPIN	61.96±0.14	26.84±0.13	28.91±0.13	37.16±0.16
DebiasGAN	62.50±0.13	27.31±0.15	29.44±0.14	37.65±0.11

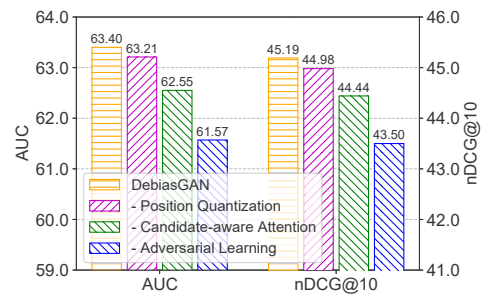
Table 2: Results of different methods on *NewsApp*.

Methods	AUC	MRR	nDCG@5	nDCG@10
EBNR	60.39±0.20	27.05±0.19	36.54±0.22	42.02±0.24
DKN	60.14±0.21	26.89±0.17	36.39±0.19	41.88±0.22
NAML	61.27±0.17	28.36±0.16	37.71±0.19	43.26±0.20
NPA	61.30±0.20	28.38±0.18	37.75±0.17	43.27±0.19
NRMS	61.52±0.16	28.65±0.14	37.92±0.15	43.46±0.17
FIM	61.67±0.14	28.74±0.15	38.03±0.16	43.58±0.18
IPW	61.96±0.14	29.11±0.13	38.34±0.16	43.80±0.17
REM	62.29±0.16	29.34±0.15	38.59±0.17	44.03±0.19
PAL	62.34±0.16	29.41±0.15	38.66±0.14	44.10±0.17
DPIN	61.98±0.15	29.14±0.13	38.36±0.14	43.82±0.16
DebiasGAN	63.40±0.14	30.52±0.12	39.80±0.16	45.19±0.15

Table 3: Results of different methods on *Uniform*.



(a) *NewsApp*.



(b) *Uniform*.

Figure 4: Effect of core components in *DebiasGAN*.

3.3 Ablation Study

We verify the effectiveness of several core techniques in our approach, including candidate-aware attention networks, adversarial learning, and position quantization. The results of *DebiasGAN* and its variants without one of these components are shown in Fig. 4. We find that both candidate-aware attention and adversarial learning can improve the model performance, especially on the

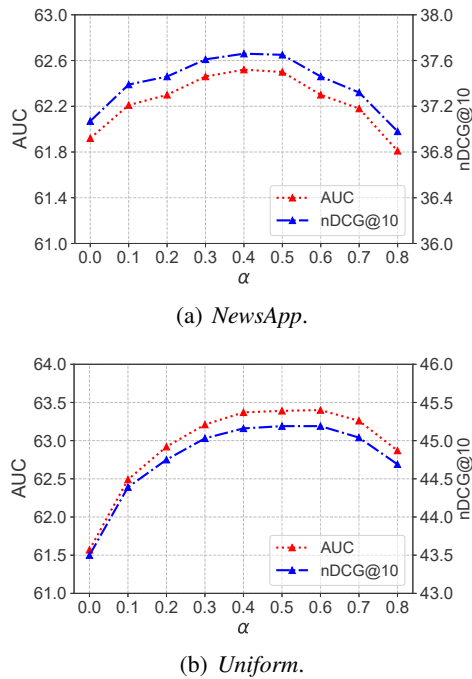


Figure 5: Influence of the loss coefficient α .

Uniform dataset. This may be because candidate-aware attention can help model the user interest in candidate news, and adversarial learning can help eliminate the effect of position biases on click prediction. In addition, quantizing the position can improve the performance. This may be because the bias effects of adjacent positions are usually similar, and quantizing positions can also reduce their sparsity to learn accurate position embeddings.

3.4 Hyperparameter Analysis

We study the impact of the loss coefficient α on the model performance. The performance in terms of AUC w.r.t. different α is shown in Fig. 5. We find that as α increases, the performance first increases and then decreases. This may be because position biases cannot be effectively removed without a sufficient intensity of adversarial gradients, while the click model may not receive adequate supervision from the main recommendation task if α is too large. Thus, we choose $\alpha = 0.5$ which yields good performance on both datasets.

4 Conclusion

In this paper, we propose a news recommendation method named *DebiasGAN* that can eliminate position biases via adversarial learning. We propose to model the interactions between user behavior and position biases to achieve personalized bias model-

ing in click prediction. In addition, we propose an adversarial debiasing method to help infer unbiased user interests in candidate news. Experiments on two real-world datasets validate that *DebiasGAN* can effectively improve news recommendation performance via position debiasing.

5 Limitation

The major limitation of the *DebiasGAN* method is the high sensitivity to the selection of the adversarial loss coefficient α , which is mainly due to the intrinsic instability of adversarial learning (Vondrick and Torralba, 2017). Thus, we plan to address this issue in our future work to help *DebiasGAN* easier to be tuned and deployed.

Acknowledgments

This work was supported by the National Key R&D Program of China 2021ZD0113902, the National Natural Science Foundation of China under Grant numbers 82090053, U1936126, and Tsinghua-Toyota Joint Research Fund 20213930033.

References

- Aman Agarwal, Kenta Takatsu, Ivan Zaitsev, and Thorsten Joachims. 2019. A general framework for counterfactual learning-to-rank. In *SIGIR*, pages 5–14.
- 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 *ACL*, pages 336–345.
- Ricardo Baeza-Yates. 2018. Bias on the web. *Communications of the ACM*, 61(6):54–61.
- Jan Panero Benway. 1999. *Banner blindness: What searching users notice and do not notice on the World Wide Web*. Rice University.
- Saumya Bhadani. 2021. Biases in recommendation system. In *Recsys*, pages 855–859.
- Jiawei Chen, Hande Dong, Yang Qiu, Xiangnan He, Xin Xin, Liang Chen, Guli Lin, and Keping Yang. 2021. Autodebias: Learning to debias for recommendation. In *SIGIR*, pages 21–30.
- Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2020. Bias and debias in recommender system: A survey and future directions. *arXiv preprint arXiv:2010.03240*.
- Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. 2008. An experimental comparison of click position-bias models. In *WSDM*, pages 87–94.

- Suyu Ge, Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2020. Graph enhanced representation learning for news recommendation. In *WWW*, pages 2863–2869.
- 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 *Recsys*, pages 452–456.
- Linmei Hu, Chen Li, Chuan Shi, Cheng Yang, and Chao Shao. 2020. Graph neural news recommendation with long-term and short-term interest modeling. *Inf. Process. & Manage.*, 57(2):102142.
- Jianqiang Huang, Ke Hu, Qingtao Tang, Mingjian Chen, Yi Qi, Jia Cheng, and Jun Lei. 2021. Deep position-wise interaction network for ctr prediction. In *SIGIR*, pages 1885–1889.
- Thorsten Joachims, Adith Swaminathan, and Tobias Schnabel. 2017. Unbiased learning-to-rank with biased feedback. In *WSDM*, pages 781–789.
- Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.
- Dugang Liu, Pengxiang Cheng, Zhenhua Dong, Xiquiang He, Weike Pan, and Zhong Ming. 2020. A general knowledge distillation framework for counterfactual recommendation via uniform data. In *SIGIR*, pages 831–840.
- Shumpei Okura, Yukihiko Tagami, Shingo Ono, and Akira Tajima. 2017. Embedding-based news recommendation for millions of users. In *KDD*, pages 1933–1942.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.
- Tao Qi, Fangzhao Wu, Chuhan Wu, and Yongfeng Huang. 2021a. Pp-rec: News recommendation with personalized user interest and time-aware news popularity. In *ACL*, pages 5457–5467.
- Tao Qi, Fangzhao Wu, Chuhan Wu, Peiru Yang, Yang Yu, Xing Xie, and Yongfeng Huang. 2021b. Hierec: Hierarchical user interest modeling for personalized news recommendation. In *ACL*, pages 5446–5456.
- Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*.
- Masahiro Sato, Sho Takemori, Janmajay Singh, and Tomoko Ohkuma. 2020. Unbiased learning for the causal effect of recommendation. In *Recsys*, pages 378–387.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.
- Carl Vondrick and Antonio Torralba. 2017. Generating the future with adversarial transformers. In *CVPR*, pages 1020–1028.
- Heyuan Wang, Fangzhao Wu, Zheng Liu, and Xing Xie. 2020. Fine-grained interest matching for neural news recommendation. In *ACL*, pages 836–845.
- Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2018a. Dkn: Deep knowledge-aware network for news recommendation. In *WWW*, pages 1835–1844.
- Xuanhui Wang, Nadav Golbandi, Michael Bendersky, Donald Metzler, and Marc Najork. 2018b. Position bias estimation for unbiased learning to rank in personal search. In *WSDM*, pages 610–618.
- Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019a. Neural news recommendation with attentive multi-view learning. In *IJCAI*, pages 3863–3869.
- Chuhan Wu, Fangzhao Wu, Mingxiao An, Jianqiang Huang, Yongfeng Huang, and Xing Xie. 2019b. Npa: Neural news recommendation with personalized attention. In *KDD*, pages 2576–2584.
- Chuhan Wu, Fangzhao Wu, Suyu Ge, Tao Qi, Yongfeng Huang, and Xing Xie. 2019c. Neural news recommendation with multi-head self-attention. In *EMNLP-IJCNLP*, pages 6390–6395.
- Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2020a. User modeling with click preference and reading satisfaction for news recommendation. In *IJCAI*, pages 3023–3029.
- Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. 2021a. Empowering news recommendation with pre-trained language models. In *SIGIR*, pages 1652–1656. ACM.
- Fangzhao Wu, Ying Qiao, Jiun-Hung Chen, Chuhan Wu, Tao Qi, Jianxun Lian, Danyang Liu, Xing Xie, Jianfeng Gao, Winnie Wu, et al. 2020b. Mind: A large-scale dataset for news recommendation. In *ACL*, pages 3597–3606.
- Xinwei Wu, Hechang Chen, Jiashu Zhao, Li He, Dawei Yin, and Yi Chang. 2021b. Unbiased learning to rank in feeds recommendation. In *WSDM*, pages 490–498.
- Jiahao Xun, Shengyu Zhang, Zhou Zhao, Jieming Zhu, Qi Zhang, Jingjie Li, Xiuqiang He, Xiaofei He, Tat-Seng Chua, and Fei Wu. 2021. Why do we click: Visual impression-aware news recommendation. In *MM*, pages 3881–3890.
- Jingwei Yi, Fangzhao Wu, Chuhan Wu, Qifei Li, Guangzhong Sun, and Xing Xie. 2021. Debiasedrec: Bias-aware user modeling and click prediction for personalized news recommendation. *arXiv preprint arXiv:2104.07360*.