Uncovering User Interest from Biased and Noised Watch Time in Video Recommendation

Haiyuan Zhao
School of Information, Renmin University of China
haiyuanzhao@ruc.edu.cn

Lei Zhang
Gaoling School of Artificial Intelligence, Renmin University of China
zhanglei1010@ruc.edu.cn

Jun Xu*
Gaoling School of Artificial Intelligence, Renmin University of China
junxu@ruc.edu.cn

Guohao Cai
Noah's Ark Lab, Huawei
caiguohao1@huawei.com

Zhenhua Dong
Noah's Ark Lab, Huawei
dongzhenhua@huawei.com

Ji-Rong Wen
Gaoling School of Artificial Intelligence, Renmin University of China
jrwen@ruc.edu.cn

ABSTRACT

In the video recommendation, watch time is commonly adopted as an indicator of user interest. However, watch time is not only influenced by the matching of users' interests but also by other factors, such as **duration bias** and **noisy watching**. Duration bias refers to the tendency for users to spend more time on videos with longer durations, regardless of their actual interest level. Noisy watching, on the other hand, describes users taking time to determine whether they like a video or not, which can result in users spending time watching videos they do not like. Consequently, the existence of duration bias and noisy watching make watch time an inadequate label for indicating user interest. Furthermore, current methods primarily address duration bias and ignore the impact of noisy watching, which may limit their effectiveness in uncovering user interest from watch time. In this study, we first analyze the generation mechanism of users' watch time from a unified causal viewpoint. Specifically, we considered the watch time as a mixture of the user's actual interest level, the duration-biased watch time, and the noisy watch time. To mitigate both the duration bias and noisy watching, we propose **D**ebiased and **D**enoised watch time **Co**rrection (D^{2}Co), which can be divided into two steps: First, we employ a duration-wise Gaussian Mixture Model plus frequency-weighted moving average for estimating the bias and noise terms; then we utilize a sensitivity-controlled correction function to separate the user interest from the watch time, which is robust to the estimation error of bias and noise terms. The experiments on two public video recommendation datasets and online A/B testing indicate the effectiveness of the proposed method.

CCS CONCEPTS

• **Information systems → Recommender systems**.

KEYWORDS

video recommendation, duration bias, noisy watching

ACM Reference Format:
Haiyuan Zhao, Lei Zhang, Jun Xu, Guohao Cai, Zhenhua Dong, and Ji-Rong Wen. 2023. Uncovering User Interest from Biased and Noised Watch Time in Video Recommendation. In *Seventeenth ACM Conference on Recommender Systems (RecSys '23), September 18–22, 2023, Singapore, Singapore.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3604915.3608797

1 INTRODUCTION

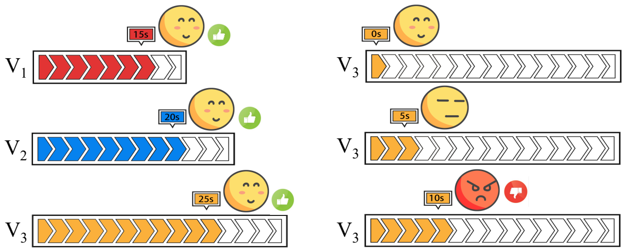
The rising of video content platforms has attracted billions of users and become more frequent in the daily use of users nowadays [7, 8, 16, 17]. In order to better satisfy the information needs of users and improve their engagement, an accurate and personalized video recommendation plays a significant role. Unlike the traditional recommendation scenario, the video recommendation adopts a streaming play pattern [9, 11]. That is, a recommender system switches to the next video and plays it automatically when the user finishes playing the previous one. This feature makes the widely used implicit feedback (e.g., user click) no longer suitable as a label to measure user interest. Compared to clicks, users' watch time indicates how much attention the user is willing to spend on this video and has been considered a better indicator of user interest [7, 24, 32, 33].

However, the length of watch time is not only determined by user interest alone but also affected by other non-interest factors. On the one hand, users tend to spend more time watching engaging videos with longer durations, resulting in longer average watch time for long videos. This phenomenon is referred to as **duration bias** [35, 39]. As shown in Fig. 1(a), all three videos v_1, v_2, v_3 are of interest to users but have different durations. It can be seen that users have a longer watch time for engaging videos with longer duration (e.g., v_3). If we regard watch time as the indicator of user interest, the duration bias will mislead the recommendation models leans to recommend more long videos. On the other hand, users need time to perceive whether they like newly recommended videos. As a result, they may watch videos they are not interested in for a while, commonly

[*] Jun Xu is the corresponding author. Work partially done at Engineering Research Center of Next-Generation Intelligent Search and Recommendation, Ministry of Education.

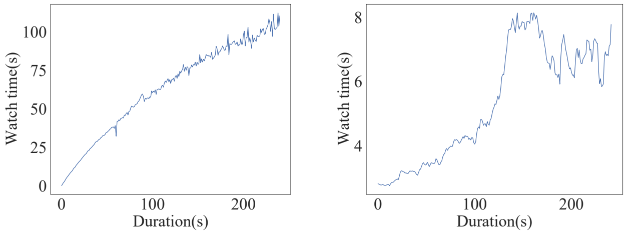
RecSys '23, September 18–22, 2023, Singapore, Singapore

ACM ISBN 979-8-4007-0241-9/23/09…$15.00
https://doi.org/10.1145/3604915.3608797



(a) Duration Bias (b) Noisy Watching

Figure 1: The illustration of duration bias and noisy watching. (a) the user watches different videos. (b) the user watches the same video.



(a) Mean watch time for videos of interest (b) Mean watch time for videos not of interest

Figure 2: The evidence of the existence of duration bias and noisy watching in the subset of the KuaiRand dataset. We calculate the mean watch time for videos that are/aren't interesting to users in different duration.

referred to as **noisy watching** [14]. Fundamentally, noisy watching results from the users' trust in the recommender system itself [1] or the clickbait content at the beginning of videos [26]. As shown in Fig. 1(b), users tend to believe that the newly recommended videos engage them when the video starts playing. Consequently, they may begin watching this video and take some time (e.g., 10s) to realize they are not interested in it. The presence of noisy watching results in users spending time watching videos they do not like, which can also mislead the recommendation models if we regard watch time as the indicator of user interest.

To verify the existence of the aforementioned duration bias and noisy watching, we conducted a pilot study on the KuaiRand dataset [9], a large-scale public video recommendation dataset collected from Kuaishou. For detecting the duration bias, we first aim to find records in the dataset that are engaging to users. Although we do not know users' latent interest behind each record, there is still some behavior feedback [38] in the dataset. Specifically, we treat one record as of interest to the user if one of the positive behavior feedback in *like*, *follow*, *forward*, *comment*, *profile enter* is presented. Then, we calculate the mean watch time in different duration on this subset. As shown in Fig. 2(a), the mean watch time of engaging videos increases with the duration growth, which verified the existence of duration bias. Similarly, for detecting noisy watching, we first regard records with negative behavior feedback like *hate* as not engaging to the user. Then, we calculate the mean watch time on this subset. As shown in Fig. 2(b), the mean watch time of videos that users aren't interested in is not zero, verifying the existence of noisy watching. Meanwhile, we can find that the curve in Fig. 2(b) increases as the duration grows. This is because longer videos usually have richer video content or a more prolonged beginning, which makes users spend more time perceiving their level of interest.

Despite the hazards, duration bias and noisy watching are much less explored as compared to many other biases in recommender system research. One heuristic way to address duration bias is to divide the watch time by the video duration, called Play Complete Rate (PCR). However, it is worth noting that the trend between watch time and duration is not a simple linear relationship according to Fig. 2. Therefore, simply dividing by duration cannot eliminate the duration bias. To better address duration bias, Zhan et al. [35] proposed to transform normal watch time prediction into duration-grouped watch time quantile to mitigate the negative effects of duration bias. Zheng et al. [39] proposed standardizing the watch time according to different video duration and leveraging the standardized score as the supervision signal to train and evaluate the video recommendation model. Although effective, there still has much space for improvement: (i) current studies only focus on addressing the duration bias while overlooking the noisy watching, which makes their predicted user interest signals still inaccurate; (ii) Existing approaches (e.g., [35] and [39]) rely on underlying assumptions (we will discuss this in section 3.3) about the distribution of user interests for correcting the duration bias. Once these assumptions are violated in practice, their performances cannot be guaranteed.

To jointly model both duration bias and noisy watching, we first conduct a causal analysis of the generation mechanism of users' watch time. Unlike current methods, which only notice the duration bias in watch time, we considered the watch time as a mixture of the user's actual interest level, the duration-biased watch time, and the noisy watch time. Then we propose a model called **D**ebiased and **D**enoised watch time **Co**rrection (D^{2}Co) to mitigate the duration bias and noisy watching. Specifically, we propose to regard the distribution of watch time in each duration length as a mixture of latent bias and noise distributions. A duration-wise Gaussian mixture model is employed to estimate the parameters of these latent distributions. Since the adjacent value of duration should have similar properties, a frequency-weighted moving average is used to smooth the estimated bias and noise parameters sequence. Then we utilized a sensitivity-controlled correction function to separate the user interest from the watch time, which is robust to the estimation error of bias and noise parameters.

Compared to existing methods, D^{2}Co enjoys the advantages of correcting the duration bias and noisy watching simultaneously in video recommendation and does not require critical assumptions on the distributions of the user interest. The major contributions of the paper include the following:

(1) We analyze the existence of duration bias and noisy watching in the video recommendation and provide a unified causal view for modeling the bias and noise simultaneously.

(2) We propose D^{2}Co, a method for mitigating both the duration bias and noisy watching. D^{2}Co can obtain user interest from watch time and does not rely on the critical assumption of user interest distribution.

(3) We conducted offline experiments on two public video recommendation datasets and an online A/B test on the real video product. Experimental results verified the effectiveness of the proposed model and theoretical conclusions.

2 RELATED WORKS

Video Recommendation With the rapid growth of video content, personalized recommendation is widely used to provide videos of interest to users in video applications. The key challenge for video recommendation is to mine user interest from various signals [23]. In a classic recommendation scenario, Click-Through-Rate (CTR) is an effective metric for measuring user interest [6, 12, 19, 22]. However, since the video recommendation scenario adopts a streaming play pattern, clicks are no longer a reliable indicator of user interest. Instead, users' watch time is commonly used as a substitute indicator of user engagement [7, 24, 32, 33]. For instance, Covington et al. [7] treated the watch time as a weight of each impressed video and utilized a weighted logistic regression for predicting watch time. Wu et al. [32] investigated the bias of watch time as well as watch percentage from an aggregated level and defined relative engagement to measure the video quality. Moreover, other trials utilized multiple user behaviors to enhance video recommendation. For example, Zhao et al. [38] proposed a large-scale multi-objective ranking system for recommending what video to watch next on an industrial video-sharing platform. Li et al. [15] designed a graph-based sequential network to simultaneously model users' dynamic and diverse interests. Wei et al. [30] considered the interactions between users and items and the item contents from various modalities.

Debiasing in Information Retrieval Alleviating the bias is of great importance in current information retrieval systems. Most efforts are devoted to address the position bias [2, 13, 34], popularity bias [29, 37, 40] and selection bias [20, 21, 27] in recent studies. Inspired by causal inference [31], a large number of debiasing methods are proposed for mitigating aforementioned biases, which includes propensity-based methods [13, 36], backdoor adjustment methods [29, 37] and causal embedding methods [4, 5, 40]. As we discussed before, the bias in video recommendation is mainly duration bias. However, only a few studies [35, 39] are focused on this bias in video recommendation. In contrast to our approach, existing methods for addressing duration bias rely on critical assumptions to achieve their unbiasedness.

Denoising in Information Retrieval To denoise data for improving model performance has been an emerging research topic in recent years. In general, the noised data is defined as the false-positive and false-negative samples among the dataset. The core idea of current studies is to mine noise data based on *Memorization Effect* [3]. That is, models can easily remember clean samples but have difficulty remembering noisy samples. For instance, Wang et al. [25] tried to mine noisy samples from the loss value and designed an adaptive threshold mechanism for truncating these samples of high loss values. Wang et al. [28] proposed to discover noisy samples from the disagreement of different models. Gao et al. [10] proposed a self-guided learning framework to collect memorized interactions at the early stage of the training. However, the above studies aim to develop a generic approach without specifically analyzing the noise in the video recommendation scenario.

3 PROBLEM STATEMENTS

3.1 Problem Formulization

The problem of video recommendation can be described as follows. Given a user u and a recalled video v with duration d, each user-video pair (u, v) is described by an n-dimensional feature vector $\mathbf{x} = \phi(u, v) \in \mathbb{R}^n$. The interest of u in v can be represented by an unobserved variable R. Without loss of generality, we assume that $R \in \{0, 1\}$ is a binary variable, which is sampled from latent Bernoulli distribution $\Pr(R = 1 \mid \mathbf{x})$. The users' watching behavior on videos can be recorded as the log data $\mathcal{D} = \{(\mathbf{x}_i, w_i, d_i)\}_{i=1}^N$, where $\mathbf{x}_i, w_i, d_i$ respectively denote the i-th user-video pair's feature vector, user's watch time on this video, and the duration of this video (e.g., in seconds)

Ideally, a scoring function $f(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ could be learned by minimizing the following ideal point-wise loss:

$$\mathcal{L}_{\text{ideal}} = \frac{1}{|\mathcal{D}|}\sum_{\mathcal{D}} -r \log\left[\sigma\left(f(\mathbf{x})\right)\right] - (1-r)\log\left[1-\sigma\left(f(\mathbf{x})\right)\right], \quad (1)$$

where r is the unobserved user's true interest in a video, σ is the sigmoid function. Equation (1) cannot be minimized because the interest indicator r is unobserved. An alternative way is naively fitting the prediction to the observed watch time w in $\mathcal{D}$:

$$\begin{aligned}\mathcal{L}_{\text{naive}} = \frac{1}{|\mathcal{D}|}\sum_{\mathcal{D}} &-\frac{w}{w_{\max}} \log\left[\sigma\left(f(\mathbf{x})\right)\right] \\ &-(1-\frac{w}{w_{\max}})\log\left[1-\sigma\left(f(\mathbf{x})\right)\right], \end{aligned} \quad (2)$$

where $w_{\max}$ is the maximum watch time in the whole $\mathcal{D}$. Note that since the value of watch time w is not between 0 and 1, it is scaled into the interval $[0, 1]$ by simply dividing with $w_{\max}$. As has been discussed, there exists a gap between the optimal solution of $\mathcal{L}_{\text{naive}}$ and that of $\mathcal{L}_{\text{ideal}}$ because the watch time w suffers from both duration bias and noisy watching. The goal of this paper is to mitigate the bias and noise, i.e., uncovering the user interest from watch time for learning better scoring function $f(\mathbf{x})$.

3.2 Causal Analysis of Watch Time

Next, we analyze how the duration bias and noisy watching affect the watch time based on the causal graph [18] shown in Figure 3. Given a user-video pair (u, v), its feature vector $\mathbf{x}$ decides both duration D and user interest R. This is reasonable because the video duration is part of the endogenous features of this video, and the level of user interest in this video can be considered as the matching extent between the user feature and the video feature. Then duration D and user interest R decide the watch time W together, as we discussed before. Since the user interest R is an unobserved variable in the dataset, watch time W is leveraged as a surrogate label of R. Unfortunately, besides the relevance R, W is also affected by the video duration D, which leads to duration bias

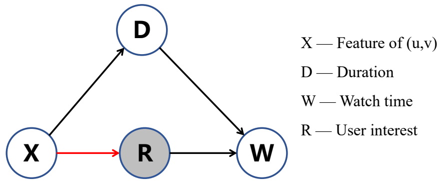


Figure 3: Causal graph of users' watch time in video recommendation. The gray node denotes the unobserved variable R. The red arrow denotes the effect that the recommendation model needs to estimate.

and noisy watching. Therefore, directly fitting watch time W will result in an erroneous video recommendation model.

According to this causal graph, we can formulate the expected watch time for a given user-video pair as follows:

$$\begin{aligned}
\mathbb{E}(W \mid \mathbf{x}) &= \sum_{W} w \Pr(W = w \mid \mathbf{x}) \\
&= \sum_{W} w \left(\sum_{D} \sum_{R \in \{0,1\}} \Pr(W = w \mid D, R) \Pr(D \mid \mathbf{x}) \Pr(R \mid \mathbf{x}) \right) \\
&= \sum_{W} w \left(\sum_{R \in \{0,1\}} \Pr(W = w \mid d, R) \Pr(R \mid \mathbf{x}) \right) \\
&= \sum_{R \in \{0,1\}} \left(\sum_{W} w \Pr(W = w \mid d, R) \right) \Pr(R \mid \mathbf{x}) \\
&= \sum_{R \in \{0,1\}} \mathbb{E}(W \mid d, R) \Pr(R \mid \mathbf{x}).
\end{aligned} \tag{3}$$

The first equation is the definition of expectation; the second equation is the decomposition of $\Pr(W = w \mid \mathbf{x})$ based on the Figure 3; the third equation is based on the fact that one video only has a unique duration and the fourth equation is the multiplication switching law. Finally, we decomposed $\mathbb{E}(W \mid \mathbf{x})$ into the mixture of $\mathbb{E}(W \mid d, R = 1)$ and $\mathbb{E}(W \mid d, R = 0)$, which is weighted by $\Pr(R = 1 \mid \mathbf{x})$ and $\Pr(R = 0 \mid \mathbf{x})$, respectively.

Specifically, the $\mathbb{E}(W \mid d, R = 1)$ represents the average time users will watch a video of duration d due to their interest, which indicates the length of duration-biased watch time. Meanwhile, the $\mathbb{E}(W \mid d, R = 0)$ represents the average time users will watch a video of duration d they are not interested in, which indicates the length of noisy watch time. $\Pr(R = 1 \mid \mathbf{x})$ indicates the user's interest level for a video. For the ease of notation, we denote $\mathbb{E}(W \mid \mathbf{x})$ as w, $\mathbb{E}(W \mid d, R = 1)$ as w_d^+, $\mathbb{E}(W \mid d, R = 0)$ as w_d^- and $\Pr(R = 1 \mid \mathbf{x})$ as $p_{\mathbf{x}}^r$ in future formulation. Then we have:

$$w = p_{\mathbf{x}}^r w_d^+ + (1 - p_{\mathbf{x}}^r) w_d^- . \tag{4}$$

Eq. (4) provides a unified formulation of duration bias and noisy watching rather than treating them as two separate mechanisms, which is beneficial for developing a unified method for addressing them simultaneously. Based on decomposition on Eq. (4), we next give the error analysis of watch time as follows:

Theorem 1 (Error of watch time). *for a given (u, v), the error between scaled watch time $\frac{w}{w_{\max}}$ and its unobserved interest probability $p_{\mathbf{x}}^r$ is:*

$$\left| \frac{w}{w_{\max}} - p_{\mathbf{x}}^r \right| = \left| \frac{w_d^+ - w_{\max}}{w_{\max}} p_{\mathbf{x}}^r + \frac{w_d^-}{w_{\max}} (1 - p_{\mathbf{x}}^r) \right|$$

$$\leq \underbrace{\frac{w_{\max} - w_d^+}{w_{\max}}}_{\textit{error of duration bias}} p_{\mathbf{x}}^r + \underbrace{\frac{w_d^-}{w_{\max}}}_{\textit{error of noisy watching}} (1 - p_{\mathbf{x}}^r).$$

The proof of the Theorem is apparent based on Eq. (4). As illustrated in Theorem 1, the upper bound of watch time's error can be divided as the linear combination of the error caused by duration bias and the error caused by noisy watching. The total error of watch time can be further reduced when both two errors are reduced. This error analysis proved the need to develop an approach to address both duration bias and noisy watching.

3.3 Analysis of Existing Methods

Methods have been proposed to address the issue brought by the duration bias, including Play Complete Rate, Watch Time Gain [39] and Duration-Deconfounded Quantile-based Method [35]. However, the noisy watching is usually overlooked in these methods. Moreover, these methods uncover users' true interests with some critical assumptions on the user interest distribution, which are not always true in the real world, as shown in the following sections.

3.3.1 *Play Complete Rate.*

In fact, the problem brought by duration bias is the different magnitude of different duration levels. In order to mitigate the effect of the magnitude, one direct idea is to scale each watch time w with its corresponding video duration d and employ this ratio as a surrogate label of user interest, which is called Play Complete Rate (PCR). For a given (u, v), its PCR is formulated as:

$$r_{\mathbf{x}}^{\mathrm{PCR}} = \frac{w}{d}. \tag{5}$$

Compared with naively adopting watch time as the indicator of user interest, PCR takes a step towards scaling the magnitude according to each duration group and achieves better results. However, it can be shown that PCR can uncover user interest from watch time if and only if $w_d^+ = C_1 d$ and $w_d^- = C_2 d$, where C_1 and C_2 are two constants. The detailed analysis can be found in Appendix A.1

In practice, the underlying assumptions of PCR can hardly be satisfied. As shown in Fig 2, the curve of w_d^+ and w_d^- with duration d is not a linear function. As a consequence, the performance of PCR cannot be guaranteed.

3.3.2 *Watch Time Gain.*

Watch time gain (WTG) [39] is a newly proposed state-of-the-art method for eliminating the duration bias. The core idea of WTG is to conduct standardization after video duration grouping, thus scaling the magnitude of watch time in each duration into the same interval. For a given (u, v), its WTG is formulated as:

$$r_{\mathbf{x}}^{\mathrm{WTG}} = \frac{w - \mu_w(d)}{\sigma_w(d)}, \tag{6}$$

where $\mu_w(d)$ is the average watch time and $\sigma_w(d)$ is the standard deviation of watch time for the videos with duration d. Different from PCR, which only considers the watch time magnitude of the current sample, WTG is a method that aims to get a relative score among each duration group, thus further reducing the influence of duration bias. However, it can be shown that WTG can uncover user interest from watch time if and only if the distribution of user interest at each duration has the same expectation and standard deviation. The detailed analysis can be found in Appendix A.2.

In fact, it is unreasonable to assume that every duration group has a consistent user interest distribution. As illustrated in Fig 3, both user interest R and video duration D are determined by the feature of (u, v). Therefore, the distribution of R and the distribution of D are still correlated, which violates the assumption of WTG.

3.3.3 Duration-Deconfounded Quantile-based Method. Duration-Deconfounded Quantile-based Method (D2Q) [35] is another state-of-the-art method for alleviating duration bias. Unlike WTG, D2Q transforms the original watch time into the quantile score in each equal-frequency duration bin. For a given (u, v), its D2Q label is formulated as:

$$r_{\mathbf{x}}^{\text{D2Q}} = \frac{|\mathcal{D}| - M\pi_m(w)}{|\mathcal{D}|}, \tag{7}$$

where $|\mathcal{D}|$ is the total number of samples in the whole dataset, M is the number of equal-frequency duration bins, $\pi_m(w) : \mathbb{R} \rightarrow \{1, 2, \cdots, \frac{|\mathcal{D}|}{M}\}$ is a descending ranking function of watch time for current bin m. Similar to WTG, D2Q is also a kind of method for obtaining relative scores among each bin group. However, it can be shown that D2Q can uncover user interest from watch time if and only if all bins have the same ranking function of user interest. See the analysis in Appendix A.3 for the details.

In order to hold the condition, it is necessary to reduce the number of bins, which in turn reduces the performance of debiasing. Moreover, the assumption is difficult to be tested in most cases.

4 OUR APPROACH: D^{2}CO

To jointly mitigate the duration bias and noisy watching and relax the above assumptions, we propose Debiased and Denoised watch time Correction (D^{2}Co). Specifically, we first employ a duration-wise Gaussian Mixture Model plus frequency-weighted moving average for estimating the bias and noise terms. Then, we utilize a sensitivity-controlled correction function to separate user interest from watch time, which can reduce the sensitivity to estimation error.

4.1 Estimating the Bias and Noise Terms

As illustrated in Eq. (4), the expected watch time w of a given (u, v) can be decomposed as the mixture of duration-biased watch time w_d^+ and noisy watch time w_d^-. From the perspective of probability, the distribution of watch time $\Pr(W = w \mid \mathbf{x})$ for a given (u, v) can also be considered as the mixture of two latent distributions: $\Pr(W = w \mid d, R = 1)$ and $\Pr(W = w \mid d, R = 0)$, which is formulated as follows:

$$\Pr(W = w \mid \mathbf{x}) = \sum_{R \in \{0,1\}} \Pr(R \mid \mathbf{x}) \Pr(W = w \mid d, R), \tag{8}$$

where $\Pr(W = w \mid d, R = 1)$ is the distribution of the watch time due to the user's interest in videos with duration d, which suffers duration bias; $\Pr(W = w \mid d, R = 0)$ is the distribution of the watch time that user watches videos with duration d they are not interested in, which is controlled by noisy watching. The weight of each component is the user interest probability $\Pr(R = 1|\mathbf{x})$ and $\Pr(R = 0|\mathbf{x})$.

To uncover user interest from the watch time, we need to estimate the parameters of the latent distributions. Here, we assume that $\Pr(W = w \mid d, R = 1)$ and $\Pr(W = w \mid d, R = 0)$ are two latent Gaussian distributions, which is a wild assumption. Then the Gaussian Mixture Model (GMM) can be utilized for estimating the parameters of latent mixture Gaussian distribution. However, Eq. (8) lies on the individual level, which means we don't have enough samples to estimate the parameters of GMM for each individual. To this end, we transform the individual-level GMM equivalently to the duration level:

$$\begin{aligned}
\Pr(W = w \mid d) &= \sum_{\mathbf{x}} \Pr(\mathbf{x} \mid d) \Pr(W = w \mid d, \mathbf{x}) \\
&= \sum_{\mathbf{x} \in \mathcal{X}_d} \Pr(\mathbf{x}) \Pr(W = w \mid \mathbf{x}) \\
&= \sum_{\mathbf{x} \in \mathcal{X}_d} \Pr(\mathbf{x}) \left(\sum_{R \in \{0,1\}} \Pr(R \mid \mathbf{x}) \Pr(W = w \mid d, R) \right) \\
&= \sum_{R \in \{0,1\}} \left(\sum_{\mathbf{x} \in \mathcal{X}_d} \Pr(\mathbf{x}) \Pr(R \mid \mathbf{x}) \right) \Pr(W = w \mid d, R).
\end{aligned} \tag{9}$$

Here, $\sum_{\mathbf{x} \in \mathcal{X}_d} \Pr(\mathbf{x}) \Pr(R \mid \mathbf{x})$ can be regarded as the average user interest in videos of duration d. We can find that the latent distributions $\Pr(W = w \mid d, R = 1)$ and $\Pr(W = w \mid d, R = 0)$ are still the same as Eq. (8). As a result, we can estimate GMM parameters at the duration-level. To verify the rationality of the adoption of duration-level GMM, we show statistics on the distribution of watch time on the KuaiRand dataset. Fig. 4(a) shows the watch time distribution of different duration groups(e.g., $Duration = 20s, 30s, 40s, 50s$). A significant bimodal phenomenon appears on those hist diagrams. However, as shown in Fig. 4(b), this bimodal phenomenon disappears if we go to the watch time distribution of duration range (e.g., $Duration < 50s$). This supports the rationality of regarding the watch time distribution as a mixture distribution in the duration-level.

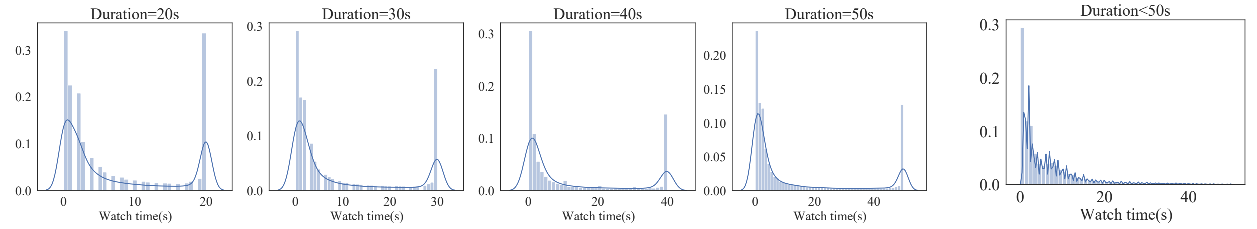
Furthermore, considering that adjacent duration should have similar duration-biased watch time and noisy watch time, we employ a bi-directional frequency-weighted moving average to smooth the estimated sequence of duration-biased watch time $\hat{w}_d^+$ and noisy watch time $\hat{w}_d^-$. That is:

$$\begin{aligned}
\widetilde{w}_{d_i}^+ &= \frac{|\mathcal{D}_{i-T}|\hat{w}_{d_{i-T}}^+ + \cdots + |\mathcal{D}_i|\hat{w}_{d_i}^+ + \cdots + |\mathcal{D}_{i+T}|\hat{w}_{d_{i+T}}^+}{|\mathcal{D}_{i-T}| + \cdots + |\mathcal{D}_i| + \cdots + |\mathcal{D}_{i+T}|}, \\
\widetilde{w}_{d_i}^- &= \frac{|\mathcal{D}_{i-T}|\hat{w}_{d_{i-T}}^- + \cdots + |\mathcal{D}_i|\hat{w}_{d_i}^- + \cdots + |\mathcal{D}_{i+T}|\hat{w}_{d_{i+T}}^-}{|\mathcal{D}_{i-T}| + \cdots + |\mathcal{D}_i| + \cdots + |\mathcal{D}_{i+T}|},
\end{aligned} \tag{10}$$

where T denotes the window size of moving average. The smoothed $\widetilde{w}_d^+$ and $\widetilde{w}_d^-$ are leveraged to separate user interest from watch time in the next section.

4.2 Separating User Interest from Watch Time

Based on Eq. (4), we can obtain the user interest with the bias term $\widetilde{w}_d^+$ and noise term $\widetilde{w}_d^-$ via affine correction, which named as



(a) Watch time dist. of different duration value

(b) Watch time dist. of duration range

Figure 4: The distribution of watch time in a different subset of KuaiRand.

$\mathrm{D^2Co(A)}$:

$$r_{\mathbf{x}}^{\mathrm{D^2Co(A)}}(w, \widetilde{w}_d^+, \widetilde{w}_d^-) = \frac{w - \widetilde{w}_d^-}{\widetilde{w}_d^+ - \widetilde{w}_d^-}. \tag{11}$$

If we estimate both the bias term and noise term accurately, Eq. (11) undoubtedly equals user interest. As shown in the following theorem:

Theorem 2 (Unbiasedness). *Given (u, v), $r_{\mathbf{x}}^{\mathrm{D^2Co(A)}}$ is unbiased if the bias and noise terms are accurately estimated:*

$$r_{\mathbf{x}}^{\mathrm{D^2Co(A)}}(w, \widetilde{w}_d^+, \widetilde{w}_d^-) \equiv p_{\mathbf{x}}^r, \quad \text{if } \widetilde{w}_d^+ = w_d^+ \wedge \widetilde{w}_d^- = w_d^-.$$

On the basis of Eq. (4), the proof of this theorem is apparent.

However, we can hardly accurately estimate the bias and noise term in practice. Once the estimation error occurs, then the above theorem will not hold. To this end, we analyzed the parameter sensitivity of $r_{\mathbf{x}}^{\mathrm{D^2Co(A)}}$ towards $\widetilde{w}_d^+$ and $\widetilde{w}_d^-$ respectively, which is given by the following theorem:

Theorem 3 (Parameter Sensitivity). *For a given disturbance (i.e., estimation error) $\delta_{\widetilde{w}_d^+}$ and $\delta_{\widetilde{w}_d^-}$ of the predict value of $\widetilde{w}_d^+$ and $\widetilde{w}_d^-$, if $w \in [\widetilde{w}_d^-, \widetilde{w}_d^+]$, the sensitivity of $r_{\mathbf{x}}^{\mathrm{D^2Co}}$ to $\widetilde{w}_d^+$ and $\widetilde{w}_d^-$ is:*

$$\mathbb{S}_{\widetilde{w}_d^+} = \left| \frac{\partial\, r_{\mathbf{x}}^{\mathrm{D^2Co}}(w, \widetilde{w}_d^+, \widetilde{w}_d^-)}{\partial\, \widetilde{w}_d^+} \delta_{\widetilde{w}_d^+} \right| = \frac{w - \widetilde{w}_d^-}{(\widetilde{w}_d^+ - \widetilde{w}_d^-)^2} \left| \delta_{\widetilde{w}_d^+} \right|,$$

$$\mathbb{S}_{\widetilde{w}_d^-} = \left| \frac{\partial\, r_{\mathbf{x}}^{\mathrm{D^2Co}}(w, \widetilde{w}_d^+, \widetilde{w}_d^-)}{\partial\, \widetilde{w}_d^-} \delta_{\widetilde{w}_d^-} \right| = \frac{\widetilde{w}_d^+ - w}{(\widetilde{w}_d^+ - \widetilde{w}_d^-)^2} \left| \delta_{\widetilde{w}_d^-} \right|,$$

where $\mathbb{S}_{\widetilde{w}_d^+}$ and $\mathbb{S}_{\widetilde{w}_d^-}$ is the sensitivity of $r_{\mathbf{x}}^{\mathrm{D^2Co}}$ to $\widetilde{w}_d^+$ and $\widetilde{w}_d^-$ respectively.

The proof of Theorem 3 is based on the definition of parameter sensitivity. This theorem indicates that the estimation error of bias and noise terms has different effects at different watch time. For $\mathbb{S}_{w_d^+}$, it has large value with the growth of w. In contrast, $\mathbb{S}_{w_d^-}$ has lower value with the growth of w. From the perspective of the entire dataset, the dataset with the majority of short watch time is mainly affected by w_d^-. In contrast, the dataset with the majority of long watch time is mainly affected by w_d^+. To this end, we proposed a sensitivity-controlled correction function that adjusts sensitivity preferences according to the proportion of watch time in the dataset:

$$r_{\mathbf{x}}^{\mathrm{D^2Co(S)}}(w, \widetilde{w}_d^+, \widetilde{w}_d^-) = \frac{\exp(\alpha w) - \exp(\alpha \widetilde{w}_d^-)}{\exp(\alpha \widetilde{w}_d^+) - \exp(\alpha \widetilde{w}_d^-)}, \tag{12}$$

where α is the sensitivity control term. We can prove that, $r_{\mathbf{x}}^{\mathrm{D^2Co(S)}}$ has a lower sensitivity to parameters w_d^+ and w_d^- compared to $r_{\mathbf{x}}^{\mathrm{D^2Co(A)}}$ through the following proposition.

Algorithm 1: The pipeline of $\mathrm{D^2Co}$

```
Input: User interactions $\mathcal{D} = \{(\mathbf{x}_i, w_i, d_i)\}_{i=1}^N$, moving
       average windows size $T$, sensitivity control term $\alpha$
$\mathbf{W}^+ \leftarrow \{\,\}, \mathbf{W}^- \leftarrow \{\,\}, \mathcal{R} \leftarrow \{\,\}$;
for $d \in \{d_{\min}, \cdots, d_{\max}\}$ do
    $\mathcal{D}' = \{(\mathbf{x}_i, w_i, d_i) \mid (\mathbf{x}_i, w_i, d_i) \in \mathcal{D} \wedge (d_i = d)\}$ ;
    $\mathbf{W}^+[d], \mathbf{W}^-[d] \leftarrow \mathrm{GMM}(\mathcal{D}', \text{components} = 2)$ ;
end
$\widetilde{\mathbf{W}}^+ \leftarrow$ Moving_Average($\mathbf{W}^+, T$) (Eq.(10));
$\widetilde{\mathbf{W}}^- \leftarrow$ Moving_Average($\mathbf{W}^-, T$) (Eq.(10)) ;
for $(\mathbf{x}_i, w_i, d_i) \in \mathcal{D}$ do
    $\mathcal{R}[i] \leftarrow r_{\mathbf{x}}^{\mathrm{D^2Co(S)}}(w_i, \widetilde{\mathbf{W}}^+[d_i], \widetilde{\mathbf{W}}^-[d_i], \alpha)$ (Eq. (12)) ;
end
return $\mathcal{R}$
```

Proposition 1 ($\mathrm{D^2Co(S)}$ has lower sensitivity). *For a given (u, v), denoting the sensitivity of $D^2Co(S)$ as $\mathbb{S}'_{w_d^+}$ and $\mathbb{S}'_{w_d^-}$, we have:*

$$\mathbb{S}'_{w_d^+} < \mathbb{S}_{w_d^+}, \quad \text{if } \alpha < 0,$$
$$\mathbb{S}'_{w_d^-} < \mathbb{S}_{w_d^-}, \quad \text{if } \alpha > 0.$$

Due to the limitation of the page, proof Proposition 1 can be found in supplementary material. In practice, we need to tune the value of α for controlling the sensitivity of $\mathrm{D^2Co(S)}$ towards w_d^+ and w_d^-.

The pipeline of our method is shown in Algorithm 1. In summary, we employ a duration-wise Gaussian Mixture Model and a frequency-weighted moving average to estimate the bias and noise terms. Then, we utilize a sensitivity-controlled correction function instead of a standard affine correction to better separate user interest from watch time. The separated user interest can be utilized as the supervision signal for learning a better recommendation model.

Table 1: Statistics of the datasets adopted in this study

Dataset	#Users	#Videos	#Interactions	Duration Ranges(s)
KuaiRand	26,988	6,598	1,266,560	[5,240]
WeChat	20,000	96,418	7,310,108	[5,60]

5 EXPERIMENTS AND RESULTS

5.1 Experimental setting

5.1.1 Datasets. For evaluating the performance of proposed D^{2}Co, we utilize two public real-world datasets: WeChat[1] and KuaiRand[2]. They are respectively collected from two large micro-video platforms, Wechat Channels and Kuaishou. We list their statistic information in Table 1. The details of these two datasets are as follows:

WeChat. This dataset is released by WeChat Big Data Challenge 2021, containing the Wechat Channels logs within two weeks. Following the practice in [39], we split the data into the first 10 days, the middle 2 days, and the last 2 days as training, validation, and test set. The adopted input features include *userid,feedid,device,authorid, bgm_song_id,bgm_singer_id,user_type*, *like*, *read_comment*, *forward*.

KuaiRand [9]. KuaiRand is a newly released sequential recommendation dataset collected from KuaiShou. As suggested in [9], we utilized one of the subsets *KuaiRand-pure* in this study. To mitigate the sparsity problem, we selected data from which the video duration is up to 4 minutes. We split the data into the first 14 days, the middle 7 days, and the last 10 days as training, validation, and test set. The adopted input features include *user_id*, *video_id*, *author_id*, *music_id*, *video_type,upload_type,tab,is_like*, *is_follow*, *is_comment*, *is_forward*, *is_profile_enter,is_hate*, *most_popular_tag*.

5.1.2 Evaluation. As we discussed before, the watch time is an unreliable label for measuring user interest. For evaluating the performance of mitigating the duration bias and noisy watching in watch time, we need to know the true user interest in the recommended video. Since the explicit feedbacks suffer the spareness problem, we cannot directly utilize them as ground truth labels in our experiments. To this end, we adopt the definition of *long_view* from the KuaiRand dataset [9] as the user interest indicator, which defines the user interest for a given (u, v) as follows:

$$r_{\mathbf{x}} = \begin{cases} 1, & \text{if } (d \leq 18s \land w = d) \lor (d > 18s \land w > 18s); \\ 0, & \text{else}; \end{cases} \tag{13}$$

It is worth noting that this kind of definition is close to Valid Viewing (VV), which is one of the online metrics we leveraged in online A/B testing (section 5.6). Unlike the RMSE used in [35] and WTG used in [39], we are mainly concerned about whether the recommendation model can rank interesting videos in top-ranking positions, so the GAUC and nDCG@k are utilized as the evaluation metric of recommendation performance.

5.1.3 Baselines. As have been described in Section 4.2, D^{2}Co has two versions **D^{2}Co(A)** and **D^{2}Co(S)**. In our experiments, we will compare our proposed method with these baselines: **PCR**, **D2Q** [35] and **WTG** [39]. To investigate the generalization of our method and baselines, we integrate them with different backbone models. Specifically, we use the classical linear recommendation model **FM** [19], the classical deep recommendation model **DeepFM** [12] and the state-of-the-art recommendation model **AutoInt** [22] as our backbone recommendation model.

Moreover, considering that the existing baselines overlook the noisy watching, we equip those baselines with denoise capability via data post-processing. Specifically, we treat all samples with less than 5 seconds of watch time as 0 values after calculating the value of baseline labels. This simple post-processing divides the noise samples by threshold so that the baselines have denoise capability, and they are denoted as **PCR-denoise**, **D2Q-denoise**, and **WTG-denoise**.

5.1.4 Implementation Details. We implement all the backbones with pytorch-fm[3], an open-source library for factorization machine models. We employ Binary Cross Entropy Loss for all baselines and our methods for fair comparisons. In particular, we transform WTG into probability via the cumulative density function $\Phi(\cdot)$ of standard Gaussian distribution. For D^{2}Co(A) and D^{2}Co(S), we clip their value into the interval $[0, 1]$. For D2Q, the group number is set to 60 in KuaiRand and 30 in WeChat. We utilize Adam as the optimizer and set the initial learning rate as 0.001. The batch size is set as 512. For all the backbone models, we set their latent embedding dimension to 10. For all methods with neural networks, the hidden units are set to 64 while the dropout ratio is set to 0.2. The value of moving average window size T is tuned in the interval $[1, 5]$, and the value of sensitivity control term α is tuned in the interval $[1e^{-2}, 5e^{-2}]$ in WeChat dataset and $[-1e^{-2}, -5e^{-2}]$ in KuaiRand dataset. We tune our hyper parameters on the validation set while evaluating the performance on the test set. The source code is available at https://github.com/hyz20/D2Co.git.

5.2 Overall Performance

Table 2 illustrates the recommendation performance of proposed D^{2}Co and other baselines. According to the result in Table 2, our proposed D^{2}Co(S) obtains the best performance on both KuaiRand and WeChat datasets and all backbones significantly. In addition, the recommendation models trained with debiased labels PCR, D2Q, and WTG outperform those trained by Watch Time by a large margin since they mitigate the duration bias. Then, Our proposed D^{2}Co(A) and D^{2}Co(S) further outperform these debias baselines since our proposed methods consider the noisy watching. Furthermore, our proposed D^{2}Co(S) has better performance than D^{2}Co(A) in both datasets. This shows the superiority of our sensitivity-controlled correction. In section 5.4, we will reveal the intrinsic reasons why D^{2}Co(S) exceeds D^{2}Co(A).

It is worth noting that those baselines equipped with denoise post-processing (PCR-denoise, D2Q-denoise, WTG-denoise) have different degrees of improvement compared to their original methods. This phenomenon clearly confirms the existence of noisy watching. However, the denoise post-processing is just a heuristic truncation of the short watch time samples, which only removes part of the noise. Hence, there still exist performance gaps between D^{2}Co(S) and most denoised baselines. Moreover, the gap between original baselines and Watch Time is larger than that between

[1] https://algo.weixin.qq.com/

[2] http://kuairand.com/

[3] https://github.com/rixwew/pytorch-fm

Table 2: The recommendation performance of D^{2}Co and other baselines in KuaiRand and WeChat. Boldface means the best performed methods (excluding Oracle), while underline means the second best performed methods, superscripts † means the significance compared to the second best performed methods with $p < 0.05$ of one-tailed t-test .

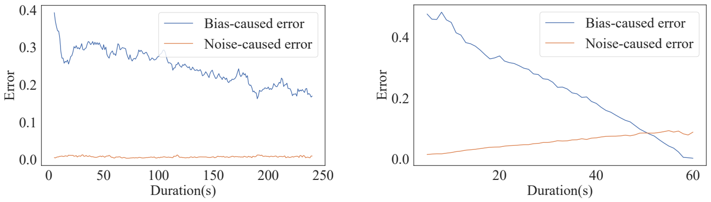
		KuaiRand				WeChat			
Backbone	Methods	GAUC	nDCG@1	nDCG@3	nDCG@5	GAUC	nDCG@1	nDCG@3	nDCG@5
FM	Watch Time	0.584	0.402	0.461	0.501	0.506	0.538	0.542	0.547
	PCR	0.626	0.432	0.482	0.517	0.532	0.557	0.560	0.565
	PCR-denoise	0.636	0.437	0.487	0.521	0.532	0.560	0.563	0.567
	D2Q	0.628	0.433	0.484	0.519	0.533	0.546	0.553	0.560
	D2Q-denoise	0.641	0.441	0.490	0.524	0.538	0.559	0.563	0.569
	WTG	0.635	0.437	0.486	0.520	0.541	0.556	0.562	0.569
	WTG-denoise	0.645	0.442	0.491	0.525	0.545	0.564	0.567	0.572
	D^{2}Co(A)	0.650	0.446	0.493	0.527	0.551	0.577	0.578	0.583
	D^{2}Co(S)	**0.653**†	**0.451**†	**0.497**†	**0.530**	**0.556**†	**0.581**†	**0.586**†	**0.590**†
	Oracle	0.664	0.456	0.502	0.535	0.556	0.585	0.587	0.590
DeepFM	Watch Time	0.593	0.402	0.464	0.503	0.506	0.554	0.555	0.560
	PCR	0.628	0.435	0.483	0.518	0.531	0.559	0.562	0.568
	PCR-denoise	0.637	0.440	0.488	0.523	0.532	0.559	0.562	0.569
	D2Q	0.635	0.437	0.489	0.522	0.532	0.550	0.554	0.562
	D2Q-denoise	0.642	0.443	0.492	0.525	0.537	0.564	0.565	0.572
	WTG	0.635	0.436	0.486	0.520	0.542	0.561	0.564	0.571
	WTG-denoise	0.647	0.444	0.493	0.526	0.544	0.571	0.570	0.577
	D^{2}Co(A)	0.653	0.447	0.496	0.528	0.551	0.574	0.576	0.583
	D^{2}Co(S)	**0.656**†	**0.451**†	**0.499**†	**0.532**	**0.555**†	**0.587**†	**0.587**†	**0.593**†
	Oracle	0.666	0.459	0.505	0.537	0.556	0.583	0.585	0.591
AutoInt	Watch Time	0.592	0.398	0.461	0.501	0.506	0.559	0.557	0.562
	PCR	0.624	0.429	0.480	0.515	0.532	0.555	0.559	0.567
	PCR-denoise	0.639	0.441	0.489	0.524	0.533	0.561	0.563	0.570
	D2Q	0.633	0.436	0.486	0.521	0.535	0.553	0.556	0.564
	D2Q-denoise	0.641	0.438	0.490	0.524	0.539	0.563	0.566	0.573
	WTG	0.637	0.437	0.487	0.521	0.544	0.562	0.563	0.570
	WTG-denoise	0.645	0.441	0.491	0.525	0.547	0.569	0.571	0.578
	D^{2}Co(A)	0.653	0.448	0.496	0.529	0.551	0.575	0.578	0.585
	D^{2}Co(S)	**0.658**†	**0.453**†	**0.499**†	**0.532**†	**0.556**†	**0.581**†	**0.586**†	**0.593**†
	Oracle	0.665	0.459	0.502	0.536	0.557	0.585	0.587	0.594

D^{2}Co(S) and original baselines in KuaiRand dataset. Therefore, we can conclude that duration bias is more harmful than noisy watching in the KuaiRand dataset. In contrast, the gap between original baselines and Watch Time is smaller than that between D^{2}Co(S) and original baselines in WeChat dataset, which indicate that noisy watching is the main problem in this dataset. We will further discuss this conclusion in section 5.3.

5.3 The Effectiveness of Mitigating Bias and Noise

Although Tabel 2 shows a significant improvement of our D^{2}Co compared to the baselines, it is still unclear how much of these improvements come from the denoise that we claim to have taken into account. In Theorem 1, we analyzed the error of watch time and divided the overall error into the duration bias-caused error and noisy watching-caused error. On this basis, we first present the curve of mean error with video duration in Fig. 5, with the estimated w_d^+ and w_d^-. In Fig. 5(a), the error caused by duration bias is much larger than that of noisy watching, and the curve of noisy watching is close to zero. This indicates that duration bias dominates the error of watch time in KuiaRand. In Fig. 5(b), the error caused by noisy watching is an increasing curve, while the error caused by duration bias is a decreasing curve. This indicates that the duration bias dominates the overall error of watch time in short-duration intervals of WeChat. However, in long-duration intervals of WeChat, the noisy watching dominates the watch time's overall error.

Then we split both KuaiRand and WeChat into three equal frequency duration ranges and evaluate the performance of each method in the corresponding subset. The results are shown in Table 3. To better reveal the performance difference, we defined the improve percentage $Imp(\%)_m = \frac{V_m - V_{wt}}{V_o - V_{wt}}$ in each subset, where V_m is the value of current method's performance, V_{wt} is the value of Watch Time's performance and V_o is the value of Oracle's performance. Actually, $Imp(\%)_m$ indicates the relative effect of debias and denoise in the current subset. For KuaiRand, although it has only duration bias caused error, our method D^{2}Co(A) and D^{2}Co(S) still exceeds the baseline, which shows the superiority of our method not relying on the critical assumptions. On WeChat, baselines and D^{2}Co(A) have similar performance in short duration while D^{2}Co(A) outperform baselines significantly in long duration. Also note that for these debiased baselines, their performances in the long duration of WeChat (e.g., $(42, 60]$) even showed declines relative to the Watch Time. As we discussed before, WeChat is affected by duration



(a) KuaiRand

(b) WeChat

Figure 5: The curve of the mean error caused by duration bias and noisy watching with the growth of duration, w.r.t KuaiRand and WeChat datasets.

Table 3: The nDCG@1 of D^{2}Co and other baselines in three equal frequency duration range. Boldface means the best performed methods (excluding Oracle), while underline means the second best performed methods, superscripts † means the significance compared to the second best performed methods with $p < 0.05$ of one-tailed t-test . The backbone recommendation model is DeepFM.

Dataset	Duration Range	Watch Time	PCR	D2Q	WTG	D^{2}Co(A)	D^{2}Co(S)	Oracle
KuaiRand	(0,32]	0.380	0.389(+32.7%)	0.391(+36.4%)	0.391(+36.8%)	**0.397(+58.5%)**†	0.397(+58.3%)	0.409
	(32,94]	0.394	0.398(+20.6%)	0.406(+67.6%)	0.402(+46.9%)	0.409(+86.9%)	**0.411(+99.3%)**	0.411
	(94,240]	0.371	0.374(+19.1%)	0.373(+10.0%)	0.375(+20.0%)	0.382(+58.6%)	**0.389(+92.9%)**†	0.390
WeChat	(0,16]	0.554	0.573(+57.0%)	0.565(+31.6%)	0.569(+44.2%)	0.579(+74.3%)	**0.591(+108.9%)**†	0.588
	(16,42]	0.549	0.555(+28.4%)	0.545(-16.1%)	0.554(+22.9%)	0.568(+85.6%)	**0.569(+91.5%)**†	0.571
	(42,60]	0.548	0.546(-20.3%)	0.544(-35.4%)	0.548(-4.8%)	0.556(+61.5%)	**0.558(+70.7%)**†	0.561

bias in short duration and noisy watching in long duration, so the results on WeChat indicate that our proposed D^{2}Co has the ability to mitigate the noisy watching, thus outperform other baselines in a large margin on the long duration videos of WeChat.

5.4 The Effectiveness of Sensitivity Control

In Theorem 3, we argue that the sensitivity of w_d^+ and w_d^- produces different hazards for different datasets, and our sensitivity control correction reduces the sensitivity by controlling the corresponding sensitive parameters in different datasets. For KuaiRand, it has many records of the long watch time. These records make the sensitivity mainly dominated by w_d^+. For WeChat, it has many records of short watch time. These records make the sensitivity mainly dominated by w_d^-. Similarly, we divide the datasets into equal-frequency groups by duration range. The larger the duration, the longer the average watch time. Fig. 6 we present the GAUC of D^{2}Co(A) and D^{2}Co(S) in different duration ranges of two datasets. As we discussed, the bottleneck of KuaiRand is those long watch time records, so our proposed D^{2}Co(S) mainly outperforms D^{2}Co(A) in a large duration range (e.g., (94,240]). Meanwhile, the bottleneck of WeChat is those short watch time records, so our proposed D^{2}Co(S) mainly outperforms D^{2}Co(A) in a small duration range (e.g., (0,16]). In general, our proposed sensitivity-controlled correction is able to control the parameter sensitivity according to the bottleneck of different datasets, thus enhancing the original D^{2}Co(A).

5.5 The Effect of Hyper-Parameters

There are two hyper-parameters of our proposed D^{2}Co. One is the size of the windows T of frequency-weighted moving average in Eq. (10). The larger the T, the smoother the bias and noise terms at adjacent times and the less specific the bias and noise terms themselves. The other is the sensitivity control term α of sensitivity-controlled correction in Eq (12). The larger the absolute value of α, the greater the decrease in sensitivity of the corresponding bias and noise parameter, but the smaller the unbiasedness of estimated user interest. In most cases, α is set to a very small value. Both T and α are essential for improving the performance of D^{2}Co. Fig. 7 illustrate the performance change of FM, DeepFM and AutoInt with different values of T and α. The figure indicates that different backbone recommendation models may have different reactions to the change of T and α. For FM (Fig. 7(a)), the best hyper-parameter is $T \in \{2, 3, 4\} \wedge \alpha = -0.07$;For DeepFM (Fig. 7(b)), the best hyper-parameter is $T \in \{2, 3, 4\} \wedge \alpha = -0.05$; For AutoInt(Fig. 7(c)), the best hyper-parameter is $T = 2 \wedge \alpha = -0.05$. In practice, it is necessary to adjust the hyper-parameters to make D^{2}Co perform best.

5.6 Online A/B Testing

We conducted online A/B testing by deploying our D^{2}Co(S) in the video feeds of Huawei browser, a platform with tens of millions of daily active users (DAU), to evaluate its effectiveness in real video recommendation products. Specifically, we randomly split the users into the control and experimental groups. For the control group, the users were served by a highly-optimized deep CTR model without training by D^{2}Co(S). For the experimental group, the users were served by the same CTR model trained with D^{2}Co(S). Tabel 4 presents the relative improvements of the base model trained with D^{2}Co(S) on five online metrics: (1) Impression Volume; (2) Valid Viewing Volume (VV); (3) Mean Watch Time (MWT); (4) Play Complete Rate (PCR); (5) Click-Through Rate (CTR). The results show that the base model training with D^{2}Co(S) consistently outperforms

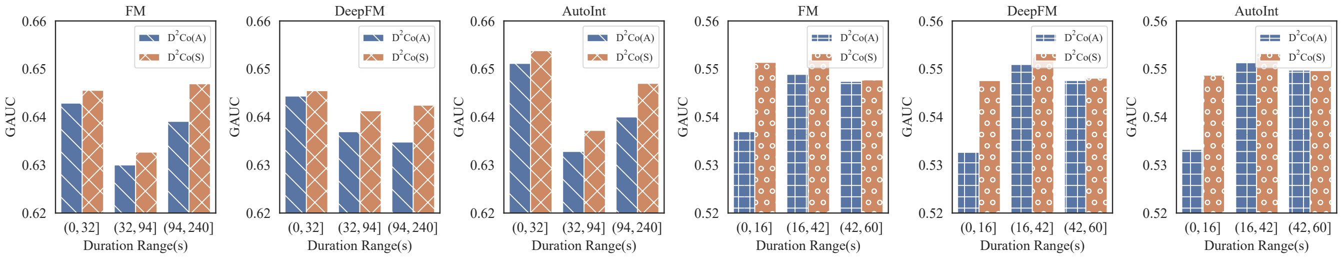
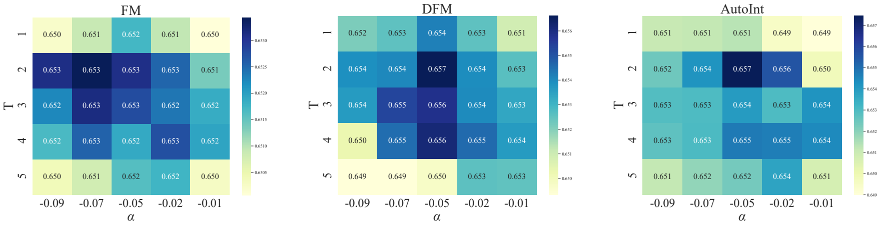


Figure 6: The effect of sensitivity control in DeCo, w.r.t different backbone models. Left three: KuaiRand; Right three: WeChat.



(a) FM (b) DeepFM (c) AutoInt

Figure 7: Hyper-parameter sensitivity of D^2Co(S) w.r.t. different backbones in KuaiRand. Each cell denotes the corresponding GAUC.

Table 4: Relative improvement (%) of D^2Co(S) to product baseline from online A/B testing

	Day1	Day2	Day3	Day4	Day5	Day6	Day7	**Average**
Impression	4.60%	6.39%	5.06%	4.49%	7.30%	5.15%	4.90%	**5.41%**
VV	7.70%	8.71%	8.19%	7.58%	11.70%	8.55%	6.04%	**8.35%**
MWT	1.91%	2.32%	1.36%	-4.00%	0.62%	-0.46%	7.43%	**1.31%**
PCR	4.72%	5.37%	3.88%	4.58%	5.09%	5.47%	4.57%	**4.81%**
CTR	2.95%	3.10%	3.00%	2.95%	4.08%	3.24%	1.08%	**2.92%**

the baseline by a large margin. One exception is the MWT, which fluctuates greatly in our A/B testing. The remarkable online improvements demonstrate the effectiveness of our proposed D^2Co in uncovering user interest from biased and noised watch time.

6 CONCLUSION

In this study, we aim to discover user interest by watch time. Due to the effect of video duration, the watch time suffers from duration bias and noisy watching simultaneously. Current methods can only address duration bias while overlooking the noisy watching. Moreover, they rely on some critical assumptions to uncover the user interest, which may not hold in practice. To this end, we propose D^2Co to mitigate both duration bias and noisy watching. Specifically, we first employ a duration-wise Gaussian Mixture Model plus frequency-weighted moving average for estimating the bias and noise terms; then, we utilize a sensitivity-controlled correction function to separate the user interest from the watch time. The experiments on two public video recommendation datasets and online A/B testing indicate the effectiveness of the proposed D^2Co.

A DETAILED ANALYSIS OF CURRENT METHODS

This section shows the detailed analysis of the assumptions for the methods in Section 3.3.

A.1 The assumption of PCR

We can further rewrite PCR as:

$$r_{\mathbf{x}}^{\text{PCR}} = \frac{w}{d} = \frac{w_d^+ p_{\mathbf{x}}^r + w_d^-(1-p_{\mathbf{x}}^r)}{d} = (\frac{w_d^+}{d} - \frac{w_d^-}{d})p_{\mathbf{x}}^r + \frac{w_d^-}{d}.$$

Then we have:

$$\forall i,j \in N, \quad (\frac{w_{d_i}^+}{d_i} - \frac{w_{d_i}^-}{d_i})p_{\mathbf{x}_\mathrm{i}}^r + \frac{w_{d_i}^-}{d_i} > (\frac{w_{d_j}^+}{d_j} - \frac{w_{d_j}^-}{d_j})p_{\mathbf{x}_\mathrm{j}}^r + \frac{w_{d_j}^-}{d_j} \Rightarrow p_{\mathbf{x}_\mathrm{i}}^r > p_{\mathbf{x}_\mathrm{j}}^r$$

$$iff. \ \frac{w_{d_i}^+}{d_i} = \frac{w_{d_j}^+}{d_j} = C_1 \wedge \frac{w_{d_i}^-}{d_i} = \frac{w_{d_j}^-}{d_j} = C_2$$

A.2 The assumption of WTG

We can further rewrite WTG as:

$$r_{\mathbf{x}}^{\mathrm{WTG}} = \frac{w - \mu_w(d)}{\sigma_w(d)} = \frac{(w_d^+ - w_d^-)p_{\mathbf{x}}^r + w_d^- - (w_d^+ - w_d^-)\mu_p(d) - w_d^-}{(w_d^+ - w_d^-)\sigma_p(d)}$$
$$= \frac{p_{\mathbf{x}}^r - \mu_p(d)}{\sigma_p(d)},$$

where $\mu_p(d)$ and $\sigma_p(d)$ are the mean user interest and standard deviation of user interest in the video group with duration d, respectively. Then we have:

$$\forall i, j \in N, \frac{p_{\mathbf{x}_i}^r - \mu_p(d_i)}{\sigma_p(d_i)} > \frac{p_{\mathbf{x}_j}^r - \mu_p(d_j)}{\sigma_p(d_j)} \Rightarrow p_{\mathrm{x_i}}^r > p_{\mathrm{x_j}}^r,$$
$$iff.\ \mu_p(d_i) = \mu_p(d_j) \wedge \sigma_p(d_i) = \sigma_p(d_j)$$

A.3 The assumption of D2Q

We can further rewrite D2Q as:

$$r_{\mathbf{x}}^{\mathrm{D2Q}} = \frac{|\mathcal{D}| - M\pi_m(w)}{|\mathcal{D}|} = \frac{\sum_k^{\frac{|\mathcal{D}|}{M}} \mathbb{I}(w > w_k)}{|\mathcal{D}|}$$
$$= \frac{\sum_k^{\frac{|\mathcal{D}|}{M}} \mathbb{I}\left((w_d^+ - w_d^-)p_{\mathbf{x}}^r + w_d^- > (w_d^+ - w_d^-)p_{\mathbf{x}_k}^r + w_d^-\right)}{|\mathcal{D}|}$$
$$= \frac{\sum_k^{|\frac{\mathcal{D}|}{M}} \mathbb{I}(p_{\mathbf{x}}^r > p_{\mathbf{x}_k}^r)}{|\mathcal{D}|} = \frac{|\mathcal{D}| - M\pi_m(p_{\mathbf{x}}^r)}{|\mathcal{D}|}$$

where $\pi_m(p_{\mathbf{x}}^r)$ is the ranking function of user interest $p_{\mathbf{x}}^r$. Then we have:

$$\forall i, j \in N, \frac{|\mathcal{D}| - M\pi_{m(i)}(p_{\mathrm{x_i}}^r)}{|\mathcal{D}|} > \frac{|\mathcal{D}| - M\pi_{m(j)}(p_{\mathrm{x_j}}^r)}{|\mathcal{D}|} \Rightarrow p_{\mathrm{x_i}}^r > p_{\mathrm{x_j}}^r,$$
$$iff.\ \pi_{m(i)}(\cdot) = \pi_{m(j)}(\cdot)$$

where $\pi_{m(i)}(\cdot)$ and $\pi_{m(j)}(\cdot)$ are the ranking function corresponding to the groups to which sample i and j belong.

ACKNOWLEDGMENTS

This work was funded by the National Key R&D Program of China (2019YFE0198200), Beijing Outstanding Young Scientist Program NO. BJJWZYJH012019100020098, Intelligent Social Governance Interdisciplinary Platform, Major Innovation & Planning Interdisciplinary Platform for the "Double-First Class" Initiative, Renmin University of China. Supported by fund for building world-class universities (disciplines) of Renmin University of China.

REFERENCES

[1] Aman Agarwal, Xuanhui Wang, Cheng Li, Michael Bendersky, and Marc Najork. 2019. Addressing Trust Bias for Unbiased Learning-to-Rank. In *The World Wide Web Conference* (San Francisco, CA, USA) *(WWW '19)*. ACM, New York, NY, USA, 4–14.

[2] Qingyao Ai, Keping Bi, Cheng Luo, Jiafeng Guo, and W. Bruce Croft. 2018. Unbiased Learning to Rank with Unbiased Propensity Estimation. In *The 41st International ACM SIGIR Conference on Research Development in Information Retrieval* (Ann Arbor, MI, USA) *(SIGIR '18)*. ACM, New York, NY, USA, 385–394. https://doi.org/10.1145/3209978.3209986

[3] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. 2017. A closer look at memorization in deep networks. In *International conference on machine learning*. PMLR, 233–242.

[4] Stephen Bonner and Flavian Vasile. 2018. Causal Embeddings for Recommendation *(RecSys '18)*. Association for Computing Machinery, New York, NY, USA, 104–112. https://doi.org/10.1145/3240323.3240360

[5] Mouxiang Chen, Chenghao Liu, Jianling Sun, and Steven C.H. Hoi. 2021. Adapting Interactional Observation Embedding for Counterfactual Learning to Rank. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 285–294. https://doi.org/10.1145/3404835.3462901

[6] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems* (Boston, MA, USA) *(DLRS 2016)*. Association for Computing Machinery, New York, NY, USA, 7–10. https://doi.org/10.1145/2988450.2988454

[7] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks for YouTube Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems* (Boston, Massachusetts, USA) *(RecSys '16)*. Association for Computing Machinery, New York, NY, USA, 191–198. https://doi.org/10.1145/2959100.2959190

[8] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, and Dasarathi Sampath. 2010. The YouTube Video Recommendation System. In *Proceedings of the Fourth ACM Conference on Recommender Systems* (Barcelona, Spain) *(RecSys '10)*. Association for Computing Machinery, New York, NY, USA, 293–296. https://doi.org/10.1145/1864708.1864770

[9] Chongming Gao, Shijun Li, Yuan Zhang, Jiawei Chen, Biao Li, Wenqiang Lei, Peng Jiang, and Xiangnan He. 2022. KuaiRand: An Unbiased Sequential Recommendation Dataset with Randomly Exposed Videos. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management* (Atlanta, GA, USA) *(CIKM '22)*. 5 pages. https://doi.org/10.1145/3511808.3557624

[10] Yunjun Gao, Yuntao Du, Yujia Hu, Lu Chen, Xinjun Zhu, Ziquan Fang, and Baihua Zheng. 2022. Self-Guided Learning to Denoise for Robust Recommendation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 1412–1422. https://doi.org/10.1145/3477495.3532059

[11] Xudong Gong, Qinlin Feng, Yuan Zhang, Jiangling Qin, Weijie Ding, Biao Li, Peng Jiang, and Kun Gai. 2022. Real-Time Short Video Recommendation on Mobile Devices. In *Proceedings of the 31st ACM International Conference on Information Knowledge Management* (Atlanta, GA, USA) *(CIKM '22)*. Association for Computing Machinery, New York, NY, USA, 3103–3112. https://doi.org/10.1145/3511808.3557065

[12] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. In *IJCAI*.

[13] Thorsten Joachims, Adith 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* (Cambridge, United Kingdom) *(WSDM '17)*. ACM, New York, NY, USA, 781–789. https://doi.org/10.1145/3018661.3018699

[14] Beibei Li, Beihong Jin, Jiageng Song, Yisong Yu, Yiyuan Zheng, and Wei Zhou. 2022. Improving Micro-Video Recommendation via Contrastive Multiple Interests. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 2377–2381. https://doi.org/10.1145/3477495.3531861

[15] Yongqi Li, Meng Liu, Jianhua Yin, Chaoran Cui, Xin-Shun Xu, and Liqiang Nie. 2019. Routing Micro-Videos via A Temporal Graph-Guided Recommendation System. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 1464–1472. https://doi.org/10.1145/3343031.3350950

[16] Shang Liu, Zhenzhong Chen, Hongyi Liu, and Xinghai Hu. 2019. User-Video Co-Attention Network for Personalized Micro-Video Recommendation. In *The World Wide Web Conference* (San Francisco, CA, USA) *(WWW '19)*. Association for Computing Machinery, New York, NY, USA, 3020–3026. https://doi.org/10.1145/3308558.3313513

[17] Yiyu Liu, Qian Liu, Yu Tian, Changping Wang, Yanan Niu, Yang Song, and Chenliang Li. 2021. Concept-Aware Denoising Graph Neural Network for Micro-Video Recommendation. In *Proceedings of the 30th ACM International Conference on Information Knowledge Management* (Virtual Event, Queensland, Australia) *(CIKM '21)*. Association for Computing Machinery, New York, NY, USA, 1099–1108. https://doi.org/10.1145/3459637.3482417

[18] Judea Pearl. 2009. *Causality*. Cambridge university press.

[19] Steffen Rendle. 2012. Factorization Machines with LibFM. *ACM Trans. Intell. Syst. Technol.* 3, 3, Article 57 (may 2012), 22 pages. https://doi.org/10.1145/2168752.2168771

[20] 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* (Houston, TX, USA) *(WSDM '20)*. Association for Computing Machinery, New York, NY, USA, 501–509. https://doi.org/10.1145/3336191.3371783

[21] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. 2016. Recommendations as treatments: Debiasing learning and evaluation. In *international conference on machine learning*. PMLR, 1670–1679.

[22] Weiping Song, Chence Shi, Zhiping Xiao, Zhijian Duan, Yewen Xu, Ming Zhang, and Jian Tang. 2019. AutoInt: Automatic Feature Interaction Learning via Self-Attentive Neural Networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (Beijing, China) *(CIKM '19)*. Association for Computing Machinery, New York, NY, USA, 1161–1170. https://doi.org/10.1145/3357384.3357925

[23] Peng Wang, Yunsheng Jiang, Chunxu Xu, and Xiaohui Xie. 2019. Overview of Content-Based Click-Through Rate Prediction Challenge for Video Recommendation *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 2593–2596. https://doi.org/10.1145/3343031.3356085

[24] Tianxin Wang, Jingwu Chen, Fuzhen Zhuang, Leyu Lin, Feng Xia, Lihuan Du, and Qing He. 2020. Capturing Attraction Distribution: Sequential Attentive Network for Dwell Time Prediction. In *ECAI 2020*. IOS Press, 529–536.

[25] Wenjie Wang, Fuli Feng, Xiangnan He, Liqiang Nie, and Tat-Seng Chua. 2021. Denoising Implicit Feedback for Recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining* (Virtual Event, Israel) *(WSDM '21)*. Association for Computing Machinery, New York, NY, USA, 373–381. https://doi.org/10.1145/3437963.3441800

[26] Wenjie Wang, Fuli Feng, Xiangnan He, Hanwang Zhang, and Tat-Seng Chua. 2021. Clicks Can Be Cheating: Counterfactual Recommendation for Mitigating Clickbait Issue. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 1288–1297. https://doi.org/10.1145/3404835.3462962

[27] 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* (Pisa, Italy) *(SIGIR '16)*. ACM, New York, NY, USA, 115–124.

[28] Yu Wang, Xin Xin, Zaiqiao Meng, Joemon M Jose, Fuli Feng, and Xiangnan He. 2022. Learning Robust Recommenders through Cross-Model Agreement. In *Proceedings of the ACM Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 2015–2025. https://doi.org/10.1145/3485447.3512202

[29] Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery Data Mining* (Virtual Event, Singapore) *(KDD '21)*. Association for Computing Machinery, New York, NY, USA, 1791–1800. https://doi.org/10.1145/3447548.3467289

[30] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019. MMGCN: Multi-Modal Graph Convolution Network for Personalized Recommendation of Micro-Video. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 1437–1445. https://doi.org/10.1145/3343031.3351034

[31] Peng Wu, Haoxuan Li, Yuhao Deng, Wenjie Hu, Quanyu Dai, Zhenhua Dong, Jie Sun, Rui Zhang, and Xiao-Hua Zhou. 2022. On the Opportunity of Causal Learning in Recommendation Systems: Foundation, Estimation, Prediction and Challenges. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*. 5646–5653. Survey Track.

[32] Siqi Wu, Marian-Andrei Rizoiu, and Lexing Xie. 2018. Beyond views: Measuring and predicting engagement in online videos. In *Twelfth international AAAI conference on web and social media*.

[33] Xing Yi, Liangjie Hong, Erheng Zhong, Nanthan Nan Liu, and Suju Rajan. 2014. Beyond Clicks: Dwell Time for Personalization *(RecSys '14)*. Association for Computing Machinery, New York, NY, USA, 113–120. https://doi.org/10.1145/2645710.2645724

[34] Bowen Yuan, Yaxu Liu, Jui-Yang Hsia, Zhenhua Dong, and Chih-Jen Lin. 2020. Unbiased Ad Click Prediction for Position-Aware Advertising Systems. In *Fourteenth ACM Conference on Recommender Systems* (Virtual Event, Brazil) *(RecSys '20)*. ACM, New York, NY, USA, 368–377. https://doi.org/10.1145/3383313.3412241

[35] Ruohan Zhan, Changhua Pei, Qiang Su, Jianfeng Wen, Xueliang Wang, Guanyu Mu, Dong Zheng, Peng Jiang, and Kun Gai. 2022. Deconfounding Duration Bias in Watch-Time Prediction for Video Recommendation. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Washington DC, USA) *(KDD '22)*. Association for Computing Machinery, New York, NY, USA, 4472–4481. https://doi.org/10.1145/3534678.3539092

[36] Xiao Zhang, Sunhao Dai, Jun Xu, Zhenhua Dong, Quanyu Dai, and Ji-Rong Wen. 2022. Counteracting user attention bias in music streaming recommendation via reward modification. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 2504–2514.

[37] Yang Zhang, Fuli Feng, Xiangnan He, Tianxin Wei, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. Causal Intervention for Leveraging Popularity Bias in Recommendation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. ACM, New York, NY, USA, 11–20. https://doi.org/10.1145/3404835.3462875

[38] Zhe Zhao, Lichan Hong, Li Wei, Jilin Chen, Aniruddh Nath, Shawn Andrews, Aditee Kumthekar, Maheswaran Sathiamoorthy, Xinyang Yi, and Ed Chi. 2019. Recommending What Video to Watch next: A Multitask Ranking System. In *Proceedings of the 13th ACM Conference on Recommender Systems* (Copenhagen, Denmark) *(RecSys '19)*. Association for Computing Machinery, New York, NY, USA, 43–51. https://doi.org/10.1145/3298689.3346997

[39] Yu Zheng, Chen Gao, Jingtao Ding, Lingling Yi, Depeng Jin, Yong Li, and Meng Wang. 2022. DVR: Micro-Video Recommendation Optimizing Watch-Time-Gain under Duration Bias. In *Proceedings of the 30th ACM International Conference on Multimedia* (Lisboa, Portugal) *(MM '22)*. Association for Computing Machinery, New York, NY, USA, 334–345. https://doi.org/10.1145/3503161.3548428

[40] Yu Zheng, Chen Gao, Xiang Li, Xiangnan He, Yong Li, and Depeng Jin. 2021. Disentangling User Interest and Conformity for Recommendation with Causal Embedding. In *Proceedings of the Web Conference 2021* (Ljubljana, Slovenia) *(WWW '21)*. ACM, New York, NY, USA, 2980–2991. https://doi.org/10.1145/3442381.3449788