Coarse-to-fine Dynamic Uplift Modeling for Real-time Video Recommendation

Chang Meng* Kuaishou Technology Beijing, China mengchang@kuaishou.com

Shuchang Liu Kuaishou Technology Beijing, China liushuchang@kuaishou.com

Xiu Li‡

Shenzhen International Graduate School, Tsinghua University Shenzhen, China li.xiu@sz.tsinghua.edu.cn

Chenhao Zhai*† Shenzhen International Graduate School, Tsinghua University Shenzhen, China dch23@mails.tsinghua.edu.cn

Xiaoqiang Feng Kuaishou Technology Beijing, China fengxiaoqiang@kuaishou.com

> Han Li‡ Kuaishou Technology Beijing, China lihan08@kuaishou.com

Xueliang Wang Kuaishou Technology Beijing, China wangxueliang03@kuaishou.com

> Lantao Hu[‡] Kuaishou Technology Beijing, China hulantao@kuaishou.com

> > Kun Gai Unaffiliated Beijing, China gai.kun@qq.com

Abstract

With the rise of short video platforms, video recommendation technology faces more complex challenges. Currently, there are multiple non-personalized modules in the video recommendation pipeline that urgently need personalized modeling techniques for improvement. Inspired by the success of uplift modeling in online marketing, we attempt to implement uplift modeling in the video recommendation scenario. However, we face two main challenges: 1) Design and utilization of treatments, and 2) Capture of user real-time interest. To address them, we design adjusting the distribution of videos with varying durations as the treatment and propose Coarse-to-fine Dynamic Uplift Modeling (CDUM) for real-time video recommendation. CDUM consists of two modules, CPM and FIC. The former module fully utilizes the offline features of users to model their long-term preferences, while the latter module leverages online real-time contextual features and request-level candidates to model users' real-time interests. These two modules work together to dynamically identify and targeting specific user groups and applying treatments effectively. Further, we conduct comprehensive experiments on the offline public and industrial datasets and online A/B test, demonstrating the superiority and effectiveness of our proposed CDUM. Our proposed CDUM is eventually fully deployed on

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

Conference acronym 'XX, June 03-05, 2018, Woodstock, NY

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM. ACM ISBN 978-1-4503-XXXX-X/18/06

https://doi.org/XXXXXXXXXXXXXXXX

the Kuaishou platform, serving hundreds of millions of users every day. The source code will be provided after the paper is accepted.

CCS Concepts

Information systems → Recommender systems.

Keywords

Uplift Modeling, Coarse-to-fine, Video Recommendation

ACM Reference Format:

Chang Meng, Chenhao Zhai, Xueliang Wang, Shuchang Liu, Xiaoqiang Feng, Lantao Hu, Xiu Li, Han Li, and Kun Gai. 2018. Coarse-to-fine Dynamic Uplift Modeling for Real-time Video Recommendation. In Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX). ACM, New York, NY, USA, 9 pages. https: //doi.org/XXXXXXXXXXXXXXXX

INTRODUCTION

The widespread popularity of short videos on social media presents new opportunities and challenges for optimizing recommendation systems on video platforms. During the video recommendation process, users show varying preferences for different types of videos through multi-faceted responses (e.g., watch time and various types of interactions). To ensure a great user experience, a common approach in the industry is to design specific modules within online recommendation pipelines to meet users' complex demands for different videos. However, most of these modules are non-personalized. For example, to provide users with a more diverse range of videos, the current online pipeline employs modules that adjust the distribution of videos with different durations. These modules categorize videos based on their durations and use uniform parameters to adjust the number of videos within different categories, which can not meet the varying preferences of different users for the duration of videos. To address such issues, we have deployed various strategy-based solutions. However, these approaches struggle to

^{*}Both authors contributed equally to this research.

[†]Work done when he was a research intern at Kuaishou Technology.

 $[\]frak{\ddagger}$ The corresponding author.

capture long-term user preferences and lack robust generalization, which hampers effective exploration and learning of user interests.

Uplift modeling, as a personalized modeling technique for populations, is used to measure the incremental impact of treatments or interventions for populations [6, 13, 17, 26]. Unlike traditional models, uplift modeling identifies the causal effect of an action by comparing treated and controlled groups, allowing for more targeted decision-making. In industrial scenarios, uplift modeling is widely applied in online marketing scenarios, aiming to provide targeted users with specific incentives, such as coupons, discounts, and bonuses. These methods are based on meta-learning models (such as T-learner [15] and S-learner [15]), tree-based models [2, 23, 33] and Deep Uplift Modeling (DUM) [13, 17-19, 30, 34]. They utilize offline user features for uplift modeling to estimate the individual treatment effect (i.e., uplift). By identifying and targeting the users that are most responsive and positively influenced by incentives, these methods significantly reduce marketing costs while improving metrics such as commission conversion rates.

Inspired by the success of uplift modeling in online marketing scenarios, we attempt to apply this approach for video recommendation. Nevertheless, as illustrated in Figure 1, there are significant differences between video recommendation scenarios and online marketing. In online marketing, the scenario relies solely on users' historical behavior data, with promotional information directly linked to products or services, making treatment and response modeling relatively straightforward. However, in the video recommendation, real-time dynamics are more pronounced. User preferences for the duration and content of the videos shift rapidly, as a variety of videos are pushed to the user (shown on the right side of Figure 1). Besides, the responses of users are also diverse and complex, thus it becomes more challenging to effectively align and associate treatments with responses. Due to the differences in scenarios, existing uplift modeling techniques cannot be seamlessly integrated or implemented, presenting the following challenges:

- Design and utilization of treatments. In the video recommendation scenarios, there are diverse and complex user responses, such as watch time, app usage duration, and video view count. These responses often constrain each other and may even exhibit inverse correlations. Therefore, it's essential to design appropriate treatments to balance these trade-offs among responses. Moreover, the thoughtful design and utilization of treatments can further aid in modeling long-term user preferences.
- Capture of user real-time interest. In video recommendation scenarios, modeling both long-term preferences and real-time interests has always been a key focus for us. The former represents the user's historical preference profile, which is more coarse-grained, while the latter captures the user's immediate interest feedback, offering a finer granularity of information. Existing uplift modeling techniques can utilize daily-level features, such as watch time over the past 7 or 14 days, to model a user's long-term preferences. However, these techniques are not capable of capturing real-time user interests effectively. On the other hand, existing methods often have biases when modeling long-term preferences. Designing effective paradigms to capture real-time interests can dynamically correct these biases, thereby enhancing the user's long-term experience.

To address these two challenges, we carefully design the treatment based on the specific characteristics of the video recommendation scenario and propose $\underline{\mathbf{C}}$ oarse-to-fine $\underline{\mathbf{D}}$ ynamic $\underline{\mathbf{U}}$ plift $\underline{\mathbf{M}}$ odeling for Real-time Video Recommendation (CDUM), which includes two modules: Coarse-grained Preference Modeling (CPM) and Finegrained Interest Capture (FIC).

To address the first challenge, we recognize that video duration distribution adjustment modules are extensively integrated across various stages of the online pipeline. By increasing the exposure ratio of short videos, it can effectively boosts commercialization metrics, while enhancing the exposure of long videos helps improve consumption metrics. This trade-off underscores the importance of optimizing the exposure ratio of videos of different durations, thereby catering to users' diverse consumption preferences and achieving more precise recommendation outcomes. Therefore, we ultimately decided to use adjusting the distribution of videos with varying durations as the treatment. To fully exploit the semantic information of the treatment, we draw inspiration from multi-interest modeling and design the CPM module with a multi-treatment learning paradigm. This module expands the treatment representation into two parts: guidance and indicator. The guidance part is used to filter and extract the information from user features, while the indicator part enhances the model's generalization across different treatments. The CPM module makes full use of users' offline features to model their long-term preferences and outputs these long-term preference scores to the FIC module.

To tackle the second challenge, we design the FIC module, which uses online real-time contextual features and request-level candidates to model users' current interests. It is mainly constructed by a MTL module, and outputs real-time interest scores for videos with different durations to finely adjust the long-term preference scores from the CPM module, dynamically refining the target users for the treatment. The incorporation and utilization of online real-time features not only correct biases in long-term preference modeling but also capture users' real-time information in a fine-grained manner. In summary, our work makes the following contributions:

- We highlight the difficulties and challenges faced in implementing uplift modeling in video recommendation scenarios, specifically regarding the design and utilization of treatments and the capture of users' real-time interests. Additionally, based on current online marketing scenarios, we have proposed insightful ideas for treatment design and utilization in the context of video recommendations.
- We have innovatively proposed Coarse-to-fine Dynamic Uplift Modeling for real-time video recommendation. This approach combines offline features with online real-time features to model users' long-term preferences and real-time interests in a coarseto-fine paradigm, dynamically identifying and targeting specific user groups and applying treatments effectively.
- We test our model on two public datasets, an industrial dataset from Kuaishou, and through online A/B experiments on the Kuaishou platform. Our model not only demonstrates superior offline performance but also achieves significant improvements in consumption metrics and user retention on Kuaishou. Our technique is ultimately deployed on the online platform, serving hundreds of millions of users.



Figure 1: A example of the difference between online marketing and video recommendation scenarios.

2 RELATED WORK

Uplift models aim to identify the target user groups for each specific treatment by accurately estimating the individual treatment effects (ITE). Existing Uplift modeling approaches fall into three categories: meta-learner based methods, tree-based methods, and neural network-based methods.

Meta-learner based methods [15, 31] are a classic framework that focuses on using existing prediction methods to learn user responses. S-learner [15] and T-learner [15] are two commonly used meta-learning paradigms. The S-learner combines the treatment variable with other features and estimates ITE through a single model. The T-learner builds models separately for the control and treatment groups and estimates ITE through the subtraction between the two models. In both paradigms, when there is a significant imbalance in the amount of data between the treatment and control groups, this imbalance may significantly affect model performance, thereby affecting the estimation of ITE. To address this issue, the X-learner [15] method has been improved based on the T-learner. It introduces the concept of propensity scores and two additional ITE estimators, combining propensity scores to weight the estimation of ITE. This method performs particularly well when there is a significant difference in the number of samples between the treatment and control groups, or when the ITE pattern is relatively simple. Tree-based methods [2, 22, 23, 33] utilize specific tree or forest structures to progressively separate sensitive subgroups corresponding to each treatment from the entire population using different metric-based splitting criteria, with treatment information included in the calculations of the splitting process.

Due to the outstanding feature extraction capabilities of deep learning, methods based on neural networks [3, 19, 26, 27, 29] have become a popular framework for uplift estimation. They introduce more complex and flexible architectures to model the response process to treatment, enabling the learning of more accurate user responses or uplift estimation values. These methods are widely applied in industries such as online marketing and precision medicine [9, 10, 18, 35]. For instance, in online marketing, these methods utilize offline user features to perform uplift modeling, and select the target population that is sensitive to incentives (e.g. coupons, discounts, and bonuses) and have a positive effect based on the estimated ITE value (i.e. uplift), which reduces marketing costs

while significantly improving indicators such as commission conversion rate. For example, EUEN [13] uses two sub-neural networks to estimate the conversion probability of the control group and the uplift effect respectively; DESCN [34] is an end-to-end multitask cross-network designed to comprehensively capture the relationship between treatment and real response while alleviating treatment bias and sample imbalance issues; EFIN [17] focuses on explicit feature interaction awareness, especially the sensitivity of non-treatment features to specific treatments, and enhances the model's robustness through an intervention constraint module.

However, all of the above methods are limited to using offline features for uplift modeling and identifying sensitive target population, which means they fail to capture users' real-time interests, thereby being unable to dynamically adjust the target population with specific treatment. Moreover, the design and utilization of treatments in these methods cannot be directly transferred to the context of video recommendations. Our proposed CDUM designs a treatment plan for the video recommendation scenario. It initially models users' long-term preferences at a coarse granularity using daily-level features, then refines the uplift modeling by capturing users' real-time interests through online real-time features, dynamically adjusting the target population for the specific treatment.

3 PRELIMINARIES

Since our designed model consists of both offline and online components, in this section, we will define and explain the problem formulation for both offline and online scenarios separately.

3.1 Offline Problem Definition

We denote $\mathcal{Z}=\{\mathbf{z_1,z_2,...,z_N}\}$ as the set of instances at the video recommendation scenarios, where $\mathbf{z}_i=(\mathbf{x}_i,\mathbf{t}_i,y_i)\in\mathcal{X}\times\mathcal{T}\times\mathcal{Y}$ and N is the number of training instances. $\mathbf{x}_i\in\mathbb{R}^{f_x\times 1}$ represents the f_x -dimensional user features and contextual features of the i-th instance. While $\mathbf{t}_i\in\mathbb{R}^{f_t\times 1}$ is the f_t -dimensional treatment features for the i-th instance. Besides, $y_i\in\{y_i^0,y_i^1,...,y_i^K\}$ is the response label under specific treatment for the i-th instance. Besides, we consider the first treatment feature of each instance to represent the treatment's index ID, with a total of K treatments, i.e., $t_i^0\in[0,K]$.

Following the Neyman-Rubin potential outcome framework [24], we denote y_i^k and y_i^0 as the potential outcome when the user in the *i*-th instance gets a particular treatment $t_i^0 \in [1, K]$ or is not treated, respectively. As we can not observe y_i^k and y_i^0 at the same time, there is no true uplift result for each instance, which is known as a key reason uplift modeling differs from traditional supervised learning. Thus, we need uplift modeling to estimate the expected individual treatment effect $\tau_k(x_i)$ for each instance.

Different from the existing uplift methods which estimates $\tau_k(x_i)$ with the calculation of the difference between $\mathbb{E}(y_i^k|t_i^0=k,x_i)$ and $\mathbb{E}(y_i^0|t_i^0=k,x_i)$, we pass $\mathbb{E}(y_i^k|t_i^0=k,x_i)$ and $\mathbb{E}(y_i^0|t_i^0=k,x_i)$ into online and combine online module (CPM) to estimate $\tau_k(x_i)$. As a result, we can dynamically refine the estimation of ITE scores and adjust the effective users under different treatments in real-time.

3.2 **Online Problem Definition**

In the online video recommendation pipeline, we have real-time contextual features and request-level candidates, which are represented as $\mathbf{s}_i = [\mathbf{s}_i^0, \mathbf{s}_i^1, ..., \mathbf{s}_i^{f_s}]$. Here $\mathbf{s}_i^j \in \mathbb{R}^{L^j \times 1}$ represents the j-th contextual features or request-level candidates for the i-th instance, and L^{j} represents the number of recent videos with which the user has interacted. It is worth noticing that we leverage a request-level instance to train the online module, i.e., modeling an instance using the features and label of each user request. On this basis, we further design $r_i^k \in \{r_i^1, ... r_i^K\}$ as the request-level label of online module. Here K corresponds to the number of treatments. To model the real-time interest of the users dynamically, r_i^k is designed to be calculated from the interaction data of exposed videos within the user candidate set at the request-level. Specifically, we divide the exposed videos into K categories based on their durations, which directly corresponds to the design of our treatment. Next, for the *k*-th duration category, we calculate the ratio of the number of long-play to short-play videos for each user and denote it as r_i^k . Additionally, we use \hat{r}_i^k as the real-time interest score, representing the user's preference for videos of different durations. This score is used to adjust the offline long-term preference score $\mathbb{E}(y_i^k|t_i^0=k,x_i)$, allowing us to estimate the user's individual treatment effect scores. Finally, we can rank leverage this score to make a rational treatment assignment.

METHOD

We propose Coarse-to-fine Dynamic Uplift Modeling for Real-time Video Recommendation (CDUM), which includes two modules: Coarse-grained Preference Modeling (CPM) and Fine-grained Interest Capture (FIC). Figure 2 illustrates the technical details of the proposed model.

Coarse-grained Preference Modeling

In this part, we introduce a module, Coarse-grained Preference Modeling, which utilizes the offline features to model the user's long-term preferences for different treatments.

4.1.1 *Feature Encoder Module.* For each instance $\mathbf{z}_i = (\mathbf{x}_i, \mathbf{t}_i, y_i)$, we encode both x_i and t_i into embeddings. For the continuous

feature, we first utilize a normalization technique to convert continuous features into sparse features. To be specific, we have:

$$x_i^{j*} = x_i^{j} / \max_i(x_i^{j}), \ t_i^{j*} = t_i^{j} / \max_i(t_i^{j}). \tag{1}$$

Then, we initialize embedding tables for both processed continuous and sparse features, and capture the embedding representations corresponding to the specific feature value. Specifically, we have:

$$\mathbf{e}_{xi} = \mathbf{E}_{x}^{T} \cdot \mathbf{x}_{i}, \ \mathbf{e}_{ti} = \mathbf{E}_{t}^{T} \cdot \mathbf{t}_{i}, \tag{2}$$

where $\mathbf{E}_x \in \mathbb{R}^{f_x \times d}$ and $\mathbf{E}_t \in \mathbb{R}^{f_t \times d}$ are the created embedding tables, f_x and f_t are the numbers of non-treatment features and treatment features, and d is the embedding size. In addition, we can get the corresponding embeddings for each feature, i.e., e_{xi} = $\{\mathbf{e}_{xi}^{0}, \mathbf{e}_{xi}^{1}, ..., \mathbf{e}_{xi}^{f_{x}}\}$ and $\mathbf{e}_{ti} = \{\mathbf{e}_{ti}^{0}, \mathbf{e}_{ti}^{1}, ..., \mathbf{e}_{ti}^{f_{t}}\}$

4.1.2 The Multi-treatment Learning Module. As we propose to utilize adjusting the distribution of videos with varying durations as treatment, the modeling of the problem becomes the paradigm of multi-treatment learning, i.e., learning the preference of users (expressed as responses) for different treatments. First, we encode the non-treatment embeddings of users with a concatenate operation, and use an average pooling operation on the treatment embedding:

$$\mathbf{e}_{xi}^* = C_{j=0}^{f_x} (\mathbf{e}_{xi}^j), \mathbf{e}_{ti}^* = \sum_{j=0}^{f_t} \mathbf{e}_{ti}^j / (f_t + 1), \tag{3}$$

Additionally, to fully leverage the treatment information, inspired by multi-interest learning [5, 16, 21], we further refine treatment embedding into indicator embedding and guidance embedding. We consider these two forms of embeddings as interest generalization under different effects of treatment. For the guidance part, we expect it to serve as a filter and information extractor for user features, expressing the guiding role of the treatment. In contrast, for the indicator part, we expect it to express the indicative role of the treatment, thereby helping the model generalize to learning tasks involving different treatments. Specifically, we first utilize an interest extraction module to refine the guidance and indicator parts from e_{ti} . Take the guidance part as an example, we have:

$$\begin{cases} \mathbf{h}_{ti}^{gui} = ReLU\left(\mathbf{e}_{ti}^* \cdot \mathbf{W}_{gui}^h + \mathbf{b}_{gui}^h\right), \\ \mathbf{e}_{ti}^{gui} = Sigmoid\left(\mathbf{h}_{ti}^{gui} \cdot \mathbf{W}_{gui}^o + \mathbf{b}_{gui}^o\right), \end{cases}$$
(4)

where $\mathbf{W}_{qui}^h \in \mathbb{R}^{d \times d_h}$ and $\mathbf{b}_{qui}^h \in \mathbb{R}^{1 \times d_h}$ are transformation matrix and bias matrix for the hidden layer, d_h is the dimension of the hidden representation \mathbf{h}_{ti}^{gui} . $\mathbf{W}_{gui}^o \in \mathbb{R}^{d_h \times d^*}$ and $\mathbf{b}_{gui}^o \in \mathbb{R}^{1 \times d^*}$ are transformation matrix and bias matrix for the output layer. \mathbf{e}_{ti}^{gui} is the guidance embedding refined from \mathbf{e}_{ti} , and d^* is the dimension of the embedding. While \mathbf{e}_{ti}^{ind} which represents the indicator embedding can be obtained in a similar process.

Furthermore, we first apply operation to e_{xi}^* similar as Equation (4), and obtain M expert representations (denoted as \mathbf{f}_m , $m \in$ $\{1, 2, ..., M\}$). After that, we utilize the guidance embedding to filter and extractor information for user features while leveraging the indicator embedding to enhance the representations of the extracted information, and have:

$$\begin{cases} \mathbf{g}_{i}^{k} = Softmax(\mathbf{W}_{g} \cdot \mathbf{e}_{ti}^{gui} + \mathbf{b}_{g}) \\ \hat{y}_{i}^{k} = h^{k} \left(\mathbf{e}_{ti}^{ind}, \sum_{m=1}^{M} \mathbf{g}_{i}^{k}(m) \cdot \mathbf{f}_{m} \right) \end{cases}$$
(5)

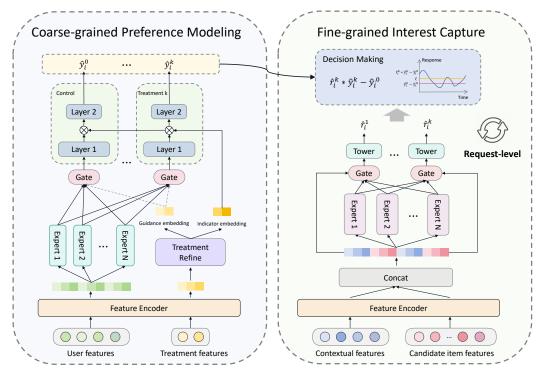


Figure 2: Illustration of the proposed CDUM framework. The light purple block on the left is the offline training part (CPM), and the light green block on the right is the online inference part (FIC).

where $\mathbf{W}_g \in \mathbb{R}^{M \times d^*}$ and $\mathbf{b}_g \in \mathbb{R}^{M \times 1}$ are feature transformation matrix and bias matrix, and $\mathbf{g}_i^k \in \mathbb{R}^{M \times 1}$ is the attention vector which is used as selector to calculate the weighted sum of all experts. $\mathbf{g}_i^k(m)$ denotes the m-th element of vector $\mathbf{g}_i^k, h^k(\cdot)$ is the tower function. And we use a similar structure with Equation (4) as the tower function here for simplicity. In addition, \mathbf{e}_{ti}^{ind} is applied as the form of mask before the final output of the tower. \hat{y}_i^k denotes the prediction score for the response of user under the k-th treatment.

4.1.3 Training Objective and Prediction. As we have obtained \hat{y}_i^k for the response of user under k-th treatment, we further use the huber loss [11] as the training loss function, i.e.,

$$\mathcal{L} = \begin{cases} 0.5 * (\hat{y}_i^k - y_i^k)^2, & if |\hat{y}_i^k - y_i^k| \le \delta \\ 0.5 * \delta^2 + \delta * (|\hat{y}_i^k - y_i^k| - \delta), & if |\hat{y}_i^k - y_i^k| > \delta \end{cases}$$
(6)

where y_i^k is the response label under k-th treatment. δ is a constant that can be adjusted dynamically. It is worth noting that in the training stage, since each sample $(\mathbf{x}_i, \mathbf{t}_i, y_i)$ is obtained by recording response only under the action of a specific treatment (e.g, k-th treatment), we only use the output \hat{y}_i^k of tower corresponding to the treatment for training, while mask operation is applied to the calculation of loss to other towers.

Finally, at the prediction stage, we can infer \hat{y}_i^k ($k \in \{0, 1, 2, ..., K\}$) for each user by replacing the feature of treatment \mathbf{t}_i iteratively. Further, we can obtain the long-term preference scores of users for different treatments, and deliver them to the next online part, FIC.

4.2 Fine-grained Interest Capture

Although we have captured the long-term preference scores, the targeted decision of the affected group still can not be adjusted dynamically in real time. In the pipeline of online video recommendation, there is rich and varied feature information, such as online real-time contextual features and request-level candidates information. Thus, we innovatively design a module, Fine-grained Interest Capture Module. This module can leverage the online features to model users' real-time interests and dynamically adjust long-term preferences accordingly, enabling real-time dynamic adjustment of the effective user groups for different treatments.

4.2.1 Module Structue. For the selection of features and labels, as mentioned in Section 3.2, we use contextual features and request-level candidates as features, i.e., s_i . At the same time, the ratio of the number of long and short broadcast exposed videos of users under each treatment classification (divided by different durations of videos) is used as the label of corresponding categories.

We first encode the online features to embeddings. This process is similar to Sectio 4.1.1, thus we get the online embeddings $\mathbf{e}_{si} = \{\mathbf{e}_{si}^0, \mathbf{e}_{si}^1, ..., \mathbf{e}_{si}^{f_s}\}$. Here $\mathbf{e}_{si}^j \in \mathbb{R}^{1 \times L^j \times d_s}, j \in [0, f_s], L^j$ represents the number of recent videos with which the user has interacted, d_s is the embedding size. Then, we apply an average pooling to the second dimension, and concatenate all the pooled embeddings together. This process can be represented as:

$$\mathbf{e}_{si}^{*} = C_{j=0}^{f_{s}} t \left(\sum_{l=1}^{L^{j}} \mathbf{e}_{si}^{j}(l) / (L^{j}) \right), \tag{7}$$

where $\mathbf{e}_{si}^{j}(l) \in \mathbb{R}^{1 \times 1 \times d_{s}}$ denotes the l-th interacted item embedding. Further, we equip \mathbf{e}_{si}^{*} with a multi-task learning network. Here, the form of the network is similar to MMOE [20], and for simplify, we denote \hat{r}_{i}^{k} ($k \in [1, K]$) as the output of each tower of this part.

4.2.2 Training Objective and Prediction. In the previous section, we have obtained the output \hat{r}_i^k for real-time interest learning of different treatments. As mentioned in Section 4.1.1, we design an online request-level label, which is the ratio of the number of longplaying videos to the number of short-playing videos. To be specific, suppose that the number of exposed videos for a user's request is *V* and we have *K* categories of treatment. The designation of the label can be formulated as:

$$\sum_{k=1}^{K} V_k = V, V_k^{long} + V_k^{short} = V_k, r_i^k = \frac{V_k^{long} + 1}{V_k^{short} + 1},$$
(8)

 $\sum_{k=1}^{K} V_k = V, V_k^{long} + V_k^{short} = V_k, r_i^k = \frac{V_k^{long} + 1}{V_k^{short} + 1},$ (8) where V_k denotes the k-th category of video duration (i.e., treatment). V_k^{long} and V_k^{short} represent the numbers of long-playing and short-playing videos, respectively.

Furthermore, for the training, we utilize the huber loss as optimization function, which is similar to the process in Section 4.1.3. While for the prediction, we deploy this module directly online, and it is invoked with every user request to calculate the real-time interest score \hat{r}_{i}^{k} .

4.2.3 Decision Making. As we have obtained the offline long-term preference score $\hat{y}_{i}^{k}(k \in \{0, 1, ..., K\})$ and online real-time interest score $\hat{r}_{i}^{k}(k \in \{1, 2, ..., K\})$, we can eventually combine them to dynamically adjust the effective user groups for different treatments. Specifically, as shown in Figure 2, we define $\hat{r}_i^k * \hat{y}_i^k - \hat{y}_i^0$ as the decision score for the k-th treatment. We then define ξ as the threshold for whether the treatment takes effect (i.e., if $\hat{r}_i^k * \hat{y}_i^k - \hat{y}_i^0 > \xi$, the k-th treatment is enabled). Thus, the multi-treatment (i.e., adjusting the distribution of videos with varying durations) for each user will be dynamically adjusted to enable or disable in real time.

OFFLINE EVALUATIONS

Experimental Setting

Dataset Description. We evaluate our proposed method on both public and real-world industrial datasets. For the public dataset, we use CRITEO [7] and LAZADA [34]. CRITEO is a dataset built by Criteo AI Labs in a real large-scale advertising scenario. It includes nearly 14 million instances, 12 features, a treatment indicator, and two labels (i.e., visit and conversion). We use visit as the target, and randomly split the dataset for the training, validation, and testing with a ratio of 8:1:1. LAZADA is a real voucher distribution business scenario dataset from the e-commerce platform Lazada. It contains 83 features, a treatment indicator, and a label. Due to the operational target strategy, the treatment allocation is selective in the actual production environment, these data containing treatment bias are utilized as training set, and a small number of other users who are not affected by the targeting strategy are used as testing set, and their treatment allocation follows the randomized controlled trial.

The industrial Kuaishou dataset consists of online user data collected 14 days prior to the launch of multiple randomized controlled trial (RCT) experiments with different treatments and nontreatments. The data features include users' inherent attributes

(region, activity, etc.), consumption information from the past 7 and 30 days (duration of usage and video views, etc.), and user's commercial attributes (including user commercial value classification). Additionally, we calculated the average app usage time during the experiment as a label. The number of instances in each group (with different treatments) is about 8,500,000. We split the data into training, validation, and test sets in a ratio of 8:1:1.

5.1.2 Evaluation Metrics. In our experiments, we adopt three widely used metrics to evaluate uplift ranking performance of different models: normalized Area Under the Uplift Curve (AUUC), normalized area under the gini curve (OINI), and uplift score at first h percentile (LIFT@h, h is set as 30). We compute these metrics using a standard python package scikit-uplift¹.

5.1.3 Baseline Models. To demonstrate the effectiveness of our proposed CDUM, we selected a set of currently popular and representative uplift modeling methods, including: S-Learner [15], T-Learner [15], BNN [12], TARNet [25], CFRNet [25], CEVAE[19], GANITE [32], DragonNet [26], FlexTENet [6], EUEN [13], DESCN [34], and EFIN [17].

For the experiments on the Kuaishou dataset, we utilize some classic multi-task models to replace the CPM model in order to test the effectiveness, including Shared Bottom [4], MMOE [20], PLE [28], MESI [8]. Since the dataset contains multiple treatments, the final evaluation metric is the average of the AUUC scores of all treatments, named AUUCava.

5.1.4 Parameter Settings. Our proposed CDUM is implemented in TensorFlow [1]. We use the Adam optimizer [14] with a learning rate of 0.001 and set the learning rate reduction factor to 0.6. The number of allowed epochs with no improvement is 2, after which the learning rate will be reduced. In addition, we set the batch size to 4096 and the embedding dimension to 32. The number of expert modules in CPM is 3. The default values of δ and ξ are 1 and 0, respectively.

Table 1: The overall performance comparison on Lazada and Cretio datasets. Boldface denotes the highest score and underline indicates the results of the best baselines. ★ represents significance level p-value < 0.05 of comparing CPM with the best baseline.

Dataset		LAZADA			CRITEO	
Metrics	QINI	AUUC	LIFT@30	QINI	AUUC	LIFT@30
S-Learner	0.0207	0.0029	0.0072	0.0902	0.0354	0.0298
T-Learner	0.0217	0.0031	0.0084	0.0790	0.0310	0.0277
BNN	0.0230	0.0033	0.0073	0.0888	0.0349	0.0286
TARNet	0.0214	0.0030	0.0072	0.0847	0.0333	0.0288
CFRNet	0.0231	0.0033	0.0087	0.0901	0.0354	0.0302
CEVAE	0.0158	0.0022	0.0074	0.0867	0.0341	0.0290
GANITE	0.0174	0.0024	0.0078	0.0818	0.0322	0.0283
DragonNet	0.0176	0.0025	0.0077	0.0851	0.0335	0.0289
FlexTENet	0.0185	0.0026	0.0079	0.0924	0.0363	0.0298
SNet	0.0238	0.0034	0.0076	0.0843	0.0331	0.0284
EUEN	0.0156	0.0022	0.0073	0.0898	0.0353	0.0300
DESCN	0.0223	0.0031	0.0075	0.0803	0.0316	0.0274
EFIN	0.0141	0.0020	0.0077	0.0859	0.0337	0.0287
CPM	0.0265*	0.0037*	0.0087*	0.0942*	0.0371*	0.0310*

¹ https://www.uplift-modeling.com/en/latest

Table 2: The overall performance comparison on Kuaishou offline dataset. Boldface denotes the highest score and underline indicates the results of the best baselines. \star represents significance level p-value < 0.05 of comparing CPM with the best baseline.

Dataset	Kuaishou	
Metrics	AUUC _{avg}	
T-Learner	0.00787	
CausalForestDML	0.00914	
Shared Bottom	0.00798	
MMOE	0.00947	
PLE	0.00998	
MESI	0.01053	
СРМ	0.01143*	

5.2 Performance Comparison

We report the evaluation results of the offline module CPM in our proposed CDUM and the baselines on two offline public datasets in Table 1. From the table, we can see that:

- On both two datasets, the performance of S-learner and T-Learner
 is not bad and even outperforms some baselines using more
 complex network architectures. This shows that in the scenario
 of large-scale uplift modeling, complex modeling methods may
 lead to suboptimal result due to the inability to quickly identify
 the sensitive features of users.
- On the LAZADA dataset, SNet performs better than other baselines overall, but on the CRITEO dataset, the best performing baseline is FlexTENet. In addition, some baselines (e.g., EUEN and DESCN) exhibit significant performance variations when evaluate on the two datasets. This inconsistency highlights that current baseline models are unable to provide reliable performance across different scenarios, lacking strong generalizability.
- Our proposed CPM performs best in all metrics of the two datasets, which demonstrates the effectiveness and applicability of our proposed offline module CPM in the task of uplift modeling.

To further validate the performance of CPM in a multi-treatment setting and to integrate it with the online module FIC for deployment, we conducted experiments on our Kuaishou dataset. Table 2 shows that CPM still maintains superior performance in the multi-treatment scenario.

Table 3: Performance of different CPM variants. ★ represents significance level p-value < 0.05 of comparing CPM with the best variants.

Dataset	LAZADA			CRITEO		
Metrics	QINI	AUUC	LIFT@30	QINI	AUUC	LIFT@30
CPM w/o indicator	0.0218	0.0030	0.0081	0.0922	0.0362	0.0308
CPM w/o guidance	0.0220	0.0031	0.0081	0.0901	0.0354	0.0304
СРМ	0.0265*	0.0037*	0.0087*	0.0942*	0.0371*	0.0310*

5.3 Ablation Study

In order to evaluate the effectiveness of the key design of the offline module CPM in our proposed method CDUM, we conduct ablation studies. Specifically, we design the following two variants:

 CPM w/o indicator: We remove the indicator embedding used to enhance the representation of extracted information. CPM w/o guidance: We use the initial feature embedding instead of the guidance embedding to generate the weights when aggregating experts in gating.

The results are shown in Table 3. It can be seen that removing any design will cause performance degradation. This demonstrates the rationality and effectiveness of our proposed module.

6 ONLINE APPLICATION

To further evaluate the online performance of our model in real product scenarios, we apply CDUM by adding a new model in the industrial recommendation service of Kuaishou, which serves billions of users. We randomly select and serve 20% of users with our model and another 20% of users with online baseline, then deploy them online from 2024-09-12 to 2024-09-29 with 18 days of online test.

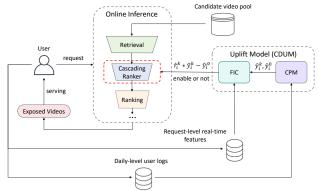


Figure 3: Illustration of the online pipeline.

6.1 System Description

As shown in Figure 3, in the pipeline of online video recommendation, a series of modules are cascaded together to filter candidate videos and finally expose the selected videos to the user. Specifically, when a user triggers a request, the corresponding user's attributes and contextual features are sent to the online service. And the online pipeline is triggered to filter and select the videos from the candidate pool in a cascading form. At last, the selected videos are exposed to the user, thus the request is over.

6.2 Scheme Overview

Our model, as illustrated in Figure 3, is developed at the bucket video duration adjustment part of Cascading Ranker. To be specific, when a user sends a request, the retrieval part first retrieves candidate videos from a candidate pool, and sends them to the Cascading Ranker. Therefore, the Cascading Ranker is triggered, and in this stage, CDUM is invoked to inference with the input of user request-level features and user's attributes, etc. In CDUM, the outputs \hat{y}_i^k, \hat{y}_i^0 of CPM are updated at a daily level, thus they are directly utilized to be calculated for the decision scores. While the FIC module makes inference based on the incoming features with the user request in real time and outputs \hat{r}_i^k . These two parts of CDUM work together to decide the enablement of treatments at the level of the user's request, thus directly acting at the bucket video duration adjustment part of the Cascading Ranker.

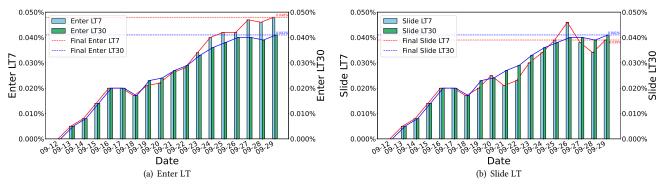


Figure 4: Online A/B test experimental results of Enter LT and Slide LT.

6.3 Online Experimental Results

6.3.1 Comparison with Online Baseline. We leverage the common consumption metrics (i.e., APP usage time and watch time) and retention metrics (i.e., Next day retention, 7-day retention, Enter LT7/LT30 and Slide LT7/LT30) in the industry to evaluate the online performance. Among them, LT7/LT30 is an unbiased estimate of medium- and long-term DAU for experimental evaluation, pointing strictly to user experience. They are key metrics that indicate user retention benefits online, and can be formulated as:

Enter LT7 =
$$\frac{\sum_{i=max(T-6,T_0)}^{T} DAU_i}{\text{Total number of active users from } T_0 \text{ to } T}$$
Slide LT7 =
$$\frac{\sum_{i=max(T-6,T_0)}^{T} DAU_i}{WAU}$$
(9)

where T stands for the date the experiment data is recorded, T_0 denotes the date that the experiment is started. DAU_i represents the number of daily active users at the i-th day. WAU represents the number of weekly active users calculated from $max(T-6, T_0)$ to T. The LT30 is calculated in a similar way. And as Kuaishou serves hundreds of millions of users every day, it is known that a slightly higher LT7/LT30 (\uparrow) at 0.01%-level is regarded significant.

Figure 4 shows the performance of LT metrics of our model compared to the online baseline during the test period. We can find that our model achieves significant improvements of **0.048**%/**0.041**% and **0.041**%/**0.039**% with respect to Enter LT7/LT30 and Silde LT7/LT30 respectively, which verifies the effectiveness of our proposed method. Besides, as shown in Table 4, our model shows significant improvement over baseline in other retention and consumption metrics, again demonstrating the superiority of CDUM. Last but not least, our model is now deployed online with full users at Kuaishou, serving for hundreds of millions of users every day.

6.3.2 Ablation Test of CDUM.. Furthermore, to demonstrate the effectiveness of the online module FIC, we conduct an ablation experiment for our proposed CDUM. To be specific, we remove the decision-making part of online adjustment, and rely solely on $\hat{y}_i^k - \hat{y}_i^0 > \xi$ ($k \in [1, K]$) to determine whether the k-th treatment is enabled or not.

As illustrated in Table 5, CDUM significantly outperforms its variant CDUM w/o FIC on consumption metrics, proving the indispensability of FIC.

Table 4: The performance of Online A/B Testing at Kuaishou platform. Boldface represents significance level *p*-value < 0.05 of comparing CDUM with the online baseline.

Dataset	Kuaishou		
Metrics	Online Baseline	CDUM	
Next day retention	-	+0.034%	
7-day retention	-	+0.038%	
APP usage time	-	+0.135%	
APP usage time (per capita)	-	+0.131%	
watch time	-	+0.188%	

Table 5: The ablation result of Online A/B Testing at Kuaishou platform. Boldface represents significance level p-value < 0.05 of comparing CDUM with the variant.

Dataset	Kuaishou		
Metrics	CDUM w/o FIC	CDUM	
APP usage time	-	+0.078%	
APP usage time (per capita)	-	+0.094%	
watch time	-	+0.129%	

7 CONCLUSIONS AND FUTURE WORK

In this paper, we explore uplift modeling in the context of video recommendation. To appropriately design treatments that capture user's real-time interests while watching videos, we propose Coarse-to-fine Dynamic Uplift Modeling (CDUM) for real-time video recommendation. This model leverages a Coarse-grained Preference Modeling (CPM) module to learn users' long-term preferences using their offline daily-level features, and then employs a Fine-grained Interest Capture (FIC) module for online inference based on real-time features, thereby capturing users' immediate interests. Extensive offline experiments on public and industrial datasets as well as online A/B test on the Kuaishou platform have demonstrated the effectiveness and superiority of our method.

For future work, we plan to explore the potential benefits of enhancing modeling at other stages of the recommendation pipeline, such as ranking, while further designing and improving the model structure and the utilization of features.

References

- [1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. {TensorFlow}: a system for {Large-Scale} machine learning. In 12th USENIX symposium on operating systems design and implementation (OSDI 16). 265–283.
- [2] Susan Athey and Guido Imbens. 2016. Recursive partitioning for heterogeneous causal effects. Proceedings of the National Academy of Sciences 113, 27 (2016), 7353-7360.
- [3] Ioana Bica, James Jordon, and Mihaela van der Schaar. 2020. Estimating the effects of continuous-valued interventions using generative adversarial networks. Advances in Neural Information Processing Systems 33 (2020), 16434–16445.
- [4] Rich Caruana. 1997. Multitask learning. Machine learning 28, 1 (1997), 41-75.
- [5] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Controllable multi-interest framework for recommendation. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2942–2951.
- [6] Alicia Curth and Mihaela Van der Schaar. 2021. On inductive biases for heterogeneous treatment effect estimation. Advances in Neural Information Processing Systems 34 (2021), 15883–15894.
- [7] Eustache Diemert, Artem Betlei, Christophe Renaudin, Massih-Reza Amini, Théophane Gregoir, and Thibaud Rahier. 2021. A large scale benchmark for individual treatment effect prediction and uplift modeling. arXiv preprint arXiv:2111.10106 (2021).
- [8] Wei Guo, Chang Meng, Enming Yuan, Zhicheng He, Huifeng Guo, Yingxue Zhang, Bo Chen, Yaochen Hu, Ruiming Tang, Xiu Li, et al. 2023. Compressed Interaction Graph based Framework for Multi-behavior Recommendation. In Proceedings of the ACM Web Conference 2023. 960–970.
- [9] Bowei He, Yunpeng Weng, Xing Tang, Ziqiang Cui, Zexu Sun, Liang Chen, Xiuqiang He, and Chen Ma. 2024. Rankability-enhanced Revenue Uplift Modeling Framework for Online Marketing. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 5093–5104.
- [10] Yinqiu Huang, Shuli Wang, Min Gao, Xue Wei, Changhao Li, Chuan Luo, Yinhua Zhu, Xiong Xiao, and Yi Luo. 2024. Entire Chain Uplift Modeling with Context-Enhanced Learning for Intelligent Marketing. In Companion Proceedings of the ACM on Web Conference 2024. 226–234.
- [11] Peter J Huber. 1992. Robust estimation of a location parameter. In Breakthroughs in statistics: Methodology and distribution. Springer, 492–518.
- [12] Fredrik Johansson, Uri Shalit, and David Sontag. 2016. Learning representations for counterfactual inference. In *International conference on machine learning*. PMLR, 3020–3029.
- [13] Wenwei Ke, Chuanren Liu, Xiangfu Shi, Yiqiao Dai, S Yu Philip, and Xiaoqiang Zhu. 2021. Addressing exposure bias in uplift modeling for large-scale online advertising. In 2021 IEEE International Conference on Data Mining (ICDM). IEEE, 1156–1161.
- [14] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).
- [15] Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. 2019. Metalearners for estimating heterogeneous treatment effects using machine learning. Proceedings of the national academy of sciences 116, 10 (2019), 4156–4165.
- [16] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. 2019. Multi-interest network with dynamic routing for recommendation at Tmall. In Proceedings of the 28th ACM international conference on information and knowledge management. 2615–2623.
- [17] Dugang Liu, Xing Tang, Han Gao, Fuyuan Lyu, and Xiuqiang He. 2023. Explicit feature interaction-aware uplift network for online marketing. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 4507–4515.
- [18] Dugang Liu, Xing Tang, Yang Qiao, Miao Liu, Zexu Sun, Xiuqiang He, and Zhong Ming. 2024. Benchmarking for Deep Uplift Modeling in Online Marketing. arXiv preprint arXiv:2406.00335 (2024).
- [19] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. 2017. Causal effect inference with deep latent-variable models. Advances in neural information processing systems 30 (2017).
- [20] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In SIGKDD. 1930–1939.
- [21] Chang Meng, Ziqi Zhao, Wei Guo, Yingxue Zhang, Haolun Wu, Chen Gao, Dong Li, Xiu Li, and Ruiming Tang. 2022. Coarse-to-Fine Knowledge-Enhanced Multi-Interest Learning Framework for Multi-Behavior Recommendation. arXiv preprint arXiv:2208.01849 (2022).
- [22] Preetam Nandy, Xiufan Yu, Wanjun Liu, Ye Tu, Kinjal Basu, and Shaunak Chatterjee. 2023. Generalized causal tree for uplift modeling. In 2023 IEEE International Conference on Big Data (BigData). IEEE, 788–798.
- [23] Nicholas J Radcliffe and Patrick D Surry. 2011. Real-world uplift modelling with significance-based uplift trees. White Paper TR-2011-1, Stochastic Solutions (2011),

- 1-33
- [24] Donald B Rubin. 2005. Causal inference using potential outcomes: Design, modeling, decisions. J. Amer. Statist. Assoc. 100, 469 (2005), 322–331.
- [25] Uri Shalit, Fredrik D Johansson, and David Sontag. 2017. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference* on machine learning. PMLR, 3076–3085.
- [26] Claudia Shi, David Blei, and Victor Veitch. 2019. Adapting neural networks for the estimation of treatment effects. Advances in neural information processing systems 32 (2019).
- [27] Zexu Sun and Xu Chen. 2024. M 3 TN: Multi-Gate Mixture-of-Experts Based Multi-Valued Treatment Network for Uplift Modeling. In ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 5065-5069.
- [28] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive layered extraction (ple): A novel multi-task learning (mtl) model for personalized recommendations. In RecSys. 269–278.
- [29] Yuxiang Wei, Zhaoxin Qiu, Yingjie Li, Yuke Sun, and Xiaoling Li. 2024. Multi-Treatment Multi-Task Uplift Modeling for Enhancing User Growth. arXiv preprint arXiv:2408.12803 (2024).
- [30] Guoqiang Xu, Cunxiang Yin, Yuchen Zhang, Yuncong Li, Yancheng He, Jing Cai, and Zhongyu Wei. 2022. Learning discriminative representation base on attention for uplift. In Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer, 200–211.
- [31] Liuyi Yao, Zhixuan Chu, Sheng Li, Yaliang Li, Jing Gao, and Aidong Zhang. 2021. A survey on causal inference. ACM Transactions on Knowledge Discovery from Data (TKDD) 15, 5 (2021), 1–46.
- [32] Jinsung Yoon, James Jordon, and Mihaela Van Der Schaar. 2018. GANITE: Estimation of individualized treatment effects using generative adversarial nets. In International conference on learning representations.
- [33] Yan Zhao, Xiao Fang, and David Simchi-Levi. 2017. Uplift modeling with multiple treatments and general response types. In *Proceedings of the 2017 SIAM International Conference on Data Mining*. SIAM, 588–596.
 [34] Kailiang Zhong, Fengtong Xiao, Yan Ren, Yaorong Liang, Wenqing Yao, Xiaofeng
- [34] Kailiang Zhong, Fengtong Xiao, Yan Ren, Yaorong Liang, Wenqing Yao, Xiaofeng Yang, and Ling Cen. 2022. DESCN: Deep entire space cross networks for individual treatment effect estimation. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 4612–4620.
- [35] Hao Zhou, Shaoming Li, Guibin Jiang, Jiaqi Zheng, and Dong Wang. 2023. Direct heterogeneous causal learning for resource allocation problems in marketing. In Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 37. 5446–5454.