2023 IEEE 39th International Conference on Data Engineering (ICDE)

Cross-Domain Disentangled Learning for E-Commerce Live Streaming Recommendation

Yixin Zhang[1,2,4*], Yong Liu[3*], Hao Xiong[4], Yi Liu[4], Fuqiang Yu[1,2], Wei He[1], Yonghui Xu[2†], Lizhen Cui[1,2†], Chunyan Miao[3,5]
[1]School of Software, Shandong University, China
[2]Joint SDU-NTU Centre for Artificial Intelligence Research (C-FAIR), Shandong University, China
[3]Alibaba-NTU Singapore Joint Research Institute, Singapore
[4]Alibaba Group, China
[5]School of Computer Science and Engineering, Nanyang Technological University, Singapore
{yixinzhang, yfq}@mail.sdu.edu.cn, {stephenliu, ascymiao}@ntu.edu.sg, {songling.xh, yunshi.ly}@alibaba-inc.com
xu.yonghui@hotmail.com, {hewei, clz}@sdu.edu.cn

2023 IEEE 39th International Conference on Data Engineering (ICDE) | 979-8-3503-2227-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICDE55515.2023.00226

***Abstract*—E-commerce live streaming as an increasingly popular sales model has generated a significant amount of gross merchandise value (GMV) for e-commerce platforms. Live streaming recommendation systems (LSRS) of e-commerce aim to recommend the most appropriate live channels for users to motivate them to buy products. Existing LSRS methods focus only on the user's interaction behaviors on the live channel (live domain) while ignoring the user's behaviors and intentions on the e-commerce product (product domain). As a result, the user's consistent purchase intentions in the cross-domain are not being fully captured, especially when user present differentiated purchase intentions in the cross-domain. How to disentangle user's consistent intentions and domain-specific intentions in the cross-domain poses a challenge to the LSRS of e-commerce platforms. In this paper, we present a live channel recommendation method, named *eLiveRec*, developed for Taobao, one of the largest e-commerce platform in the world. Specifically, eLiveRec employs the disentangled encoder module to learn user's cross-domain consistent intentions and domain-specific intentions. Then, an adaptive multi-task learning framework is developed to jointly optimize the multiple objectives (*e.g.*, stay time, click goods bag, and click products after entering channel) related to live streaming recommendation. In this way, the performance of live streaming recommendation can be further improved and conform to standard industry RS paradigms. Extensive experiments are conducted on a large-scale industry dataset collected from Taobao Live platform have been performed. Both online and offline experimental results indicate that eLiveRec consistently outperforms existing state-of-the-art baseline methods.**

***Index Terms*—E-commerce, Live streaming, Recommendation systems**

I. Introduction

The improvement of network bandwidth and the popularity of portable intelligent mobile devices motivate the emergence of live streaming, a new form of entertainment that has occupied most of our fragmented time. Representative live streaming platforms include TikTok[1], YouTube Live[2], and Taobao Live[3], which have tens of millions of daily active users. As an emerging media, live streaming has brought many innovations and new forms of business, *e.g.*, e-commerce live streaming. According to the survey report[4], 840 million Chinese people shopped online in the past year; 700 million people, or 68.2 percent of the population, watched the live streaming. Among them, there are 460 million users of e-commerce live streaming domain, accounting for 44.9 percent of the total netizens, with an annual growth rate of 19.5 percent.

Fig. 1: A glance of Taobao online live streaming scene.

With the emergence of e-commerce live streaming, many traditional businesses also explore a "live streaming + shopping" model. As shown in Fig. 1, the homepage has a series of live channels, each of which can be mounted with a goods bag, *i.e.*, a collection of products to be sold. Most of the streaming contents in the channel are explaining the mounted products. Meanwhile, the user may interact with product/anchor after entering the channel.

Live streaming has a profound impact on the daily life of the public with its advantages of strong real-time performance, high participation rate, and fast transmission speed. As a consequence, some recent studies [1]–[3] focus on the recommendation of live streaming and shed fresh insight into this emerging field. Furthermore, E-commerce live streaming provides viewers with richer interactive experience in the

∗ These two authors contributed equally to this work.
† Corresponding author

[1]http://www.tiktok.com
[2]https://www.youtube.com/live/
[3]https://taobaolive.taobao.com/
[4]https://report.iresearch.cn/report/202207/4029.shtml

2375-026X/23/$31.00 ©2023 IEEE
DOI 10.1109/ICDE55515.2023.00226

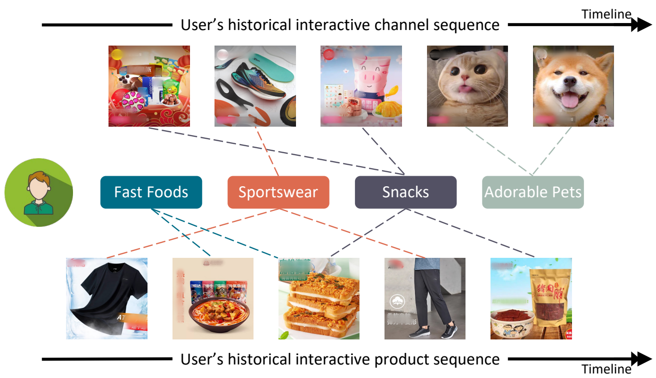

Fig. 2: Illustration of consistent and domain-specific purchase intent presented by user in the cross-domains.

virtual shopping scene, experiencing unprecedented growth, especially during the COVID-19 pandemic period. Nowadays, "live streaming + shopping" has become the pillar business of the Taobao platform. Unfortunately, there is little specific research has been conducted on e-commerce live streaming. Which is more challenging for balancing the entertainment attributes with motivating users to purchase products than LSRS. Exactly, the distinctive characteristics of the scene also raise the following challenges:

- User's intentions are complex and inconsistent across two domains. For example, in Fig. 2, we show an example about the user's behaviors and potential intentions. Specifically, the user may have consistent intentions in both product and live domains (*i.e.*, *Sportwear* and *Snacks*), while retaining specific intentions in each domain (*i.e.*, *Fast Foods* in product domain, *Adorable Pets* in live domain). Due to the significant differences between them, most existing methods fail to disentangle user's consistent intentions and domain-specific intentions.
- Directly modeling the multiple objectives related to live streaming recommendation is difficult. From Fig. 1, we can observe that the interactions between users and channels/products have a naive hierarchical structure. Firstly, the user watches a certain channel in the candidate pooling, which is called intra-channel behavior. Meanwhile, the user clicks the products or goods bag mounted in the channel, which behaviors are called inter-channel behaviors. Inter-channel behavior, which is related to the current channel, plays a variable role in the recommendation. However, the process of heuristically exploring the correlation degree of different behaviors with no pattern to follow, which only achieves sub-optimal performance and even brings negative effects.

In this paper, we propose a novel e-commerce live streaming recommendation model, named **eLiveRec**, which mainly focuses on solving the aforementioned challenges. Specifically, eLiveRec utilizes disentangled representation learning to model the user's consistent intentions and her domain-specific intentions in both product and live domains. This helps effectively exploit user's behavior data in product domain to improve the performance of live streaming recommendation. Moreover, inter-channel behavior predictions are considered as auxiliary tasks while intra-channel behavior prediction as the primary task. Auxiliary tasks are adaptive accommodated by a weighting network to quantify the consistency of each auxiliary task with respect to the primary task.

The main contributions made in this work are as follows:

- To the best of our knowledge, this is the first study investigating e-commerce live streaming recommendation, which have a positive impact on both community development and business value.
- We propose to disentangle user's consistent intentions and domain-specific intentions. To that end, we design a disentangled encoder, which can effectively disentangle the representations through introduced regularization terms.
- An adaptive multi-task learning framework is also designed to jointly optimize user's intra-channel behavior, as well as her inter-channel behavior.
- We have performed extensive experiments on a large-scale dataset collected from Taobao Live streaming platform. Both online and offline experimental results demonstrate that the proposed eLiveRec method consistently outperforms state-of-the-art baseline methods.

II. Related Work

In this section, we review the most relevant existing works about live streaming recommendation, cross-domain recommendation, and multi-task recommendation.

A. Live Streaming Recommendation

Recently, live streaming, as a newly emerging social media, has already attracted researchers' attention to this spot. Some recent work focus on the recommendation task by analyzing user-channel interaction patterns. [2] studies recommendation in this setting of a dynamically evolving set of available items. [3] aims to simultaneously perform prediction through a bi-directional prediction framework from two sides, viewer and anchor. [1] captures the matching of the anchor's and viewer's preferences and extract the related features for their representations. [4] combines probabilistic matrix factorization model with deep learning model to extract useful features from interactions for watching duration prediction. To sum up, the mainstream methods focus on learning user representations and making recommendations through sequential modeling.

As an online service deployed in the ranking phase, we perpetuate the paradigm of the CTR prediction task. One stream is about modeling the high-order interactions between different features. Wide&Deep [5] is the first deep learning method to model explicit and implicit feature interactions. Furthermore, DeepFM [6] replaces the wide net with FM [7]. After that, a series of works [8], [9] are proposed to model the feature interactions more effectively. The other stream of models focus on mining sequential patterns from consecutive user behaviors. DIN [10] is the representative work that leverages target attention network for user's interest modeling.

As the successor, DIEN [11] aims to capture evolving user interest. CAN [12] explores the potential of feature co-action based on DIEN. BST [13] and ETA [14] propose to use multi-head attention mechanism to learn a deeper representation. Other various models [15], [16] are also proposed.

Nonetheless, none of the work is concentrated on the e-commerce live streaming field, which there are inevitable interactions between users, live channels and products. How to tackle new challenges for the new scene (*i.e.*, inconsistent intentions across domains and hierarchical behavior structure), that is the focus of our work as opposed to existing methods.

B. Cross-Domain Recommendation

Cross-domain recommendation systems [17] have emerged to utilise the relatively richer information, *i.e.*, user/item information, reviews, tags and observed ratings, from the richer (source) domain to improve the recommendation accuracy in the sparser (target) domain.

Some popular methods are often based on transfer learning techniques to transfer the user/item embeddings from the source domain for providing enhanced recommendations, which includ CCCFNet [18], CoNet [19], DDTCDR [20] and etc. These methods assume that auxiliary user/item interactive behaviors across different domains will benefit the user modeling of target domain. Meanwhile, several recent studies MiNet [21], DASL [22] aim to address cross-domain sequential recommendation task by modeling user intention from consecutive behaviors across domains.

While these cross-domain sequential recommendation models achieve significantly better performance over the classical models, they mainly focus on modeling sequential patterns across domains, without considering the gaps between different domains as well as inconsistent intentions. Therefore, in this paper, we model user's consistent intentions and domain-specific intentions in the cross-domain setting by adopting the disentangled representation learning paradigm, which has applied on various fields, *e.g.*, computer vision [23], natural language processing [24], recommendation system [25]–[27].

C. Multi-Task Recommendation

In e-commerce recommendation scenes, addressing certain business proposal may lead to various optimization objectives. A common strategy is to synchronously train primary tasks, CTR prediction in most cases, and the related auxiliary tasks of different objectives. Among them, some typical multi-task learning models [28], [29] aim to improve the performance across all tasks by designing more effective shared-bottom module structure. Recent works [30], [31] propose to balance the joint learning of all tasks through adaptive strategy to avoid the situation where one or more tasks have a dominant influence on the shared-bottom network. In the LSRS of E-commerce, other than the classical CTR metric, several important second-hop metrics are also used to evaluate the effectiveness of the method, *e.g.*, stay time. More specifically, the inter-channel behaviors occur only after the user enters the live channel, which is different from traditional multi-tasks. In response to the above, we proposed a weighting network to learn adaptive weights for different auxiliary tasks to address the challenge of hierarchical behaviors, which related with live streaming recommendation.

III. Preliminaries

The top priority of personalized e-commerce live streaming recommendation is to estimate the probability that a viewer would like to watch a specific live channel. Furthermore, we estimate the probability of user interaction in the specific channel. We use user's recent behaviors to help model her long-term dynamic preferences. In this work, a user's recent behaviors include her recent clicked products and recent watched live channels. Moreover, the real-time atmosphere of the live channel affects the user's real-time preferences in the certain channel, which leads to the interactive behaviors in this channel. All behaviors include staying, and commenting, liking, following the channel anchor, and purchasing, clicking mounted products, etc.

We denote the target live channel by v_t, the i-th interaction behavior in the target channel by a_{i,v_t}, the user's product clicked sequence and live channel watched sequence by $S = \{s_1, s_2, \cdots, s_n\}$ and $V = \{v_1, v_2, \cdots, v_l\}$, respectively, where n and l denote the length of each sequence. The live channel prediction task can then be formalized as follows,

$$p(v_t|S, V) \sim f_{intra}(S, V, v_t), \tag{1}$$

where the user is represented by her recently interacted products S and live channels V, and $p(v_t|S, V)$ is the probability that the user would like to watch the target live channel v_t . Moreover, f_{intra} is the module function of **Intra-channel Behavior Modeling** used to estimate $p(v_t|S, V)$.

The interaction behaviors prediction task in the target channel can also be formalized as follows,

$$p(a_i|S, V, v_t) \sim f_{inter}(a_i, S, V, v_t), \tag{2}$$

where a_i is the i-th behavior. Here, we only keep *stay time*, *click goods bag* and *click products* three behaviors for simplicity. And $p(a_i|S, V, v_t)$ is the probability that the user performs interaction a_i within the target channel v_t. Similarly, f_{inter} is the module function of **Inter-channel Behavior Modeling** used to estimate the probability.

IV. The proposed eLiveRec model

Fig. 3 shows the overall structure of the proposed eLiveRec model. Next, we introduce the main components of eLiveRec in details.

A. Embedding Initialization

Data instance is a crucial element of real-world recommender system. Each instance is typically presents a multi-filed form, including *user profile*, *spatiotemporal context*, *clicked products sequence*, *watched channels sequence*, *target channel*, *target channel real-time context*, *label*. Each field is composed of multiple characteristics: *user profile* contains *user id*, *nickname* and so on; *spatiotemporal context* contains *time*,

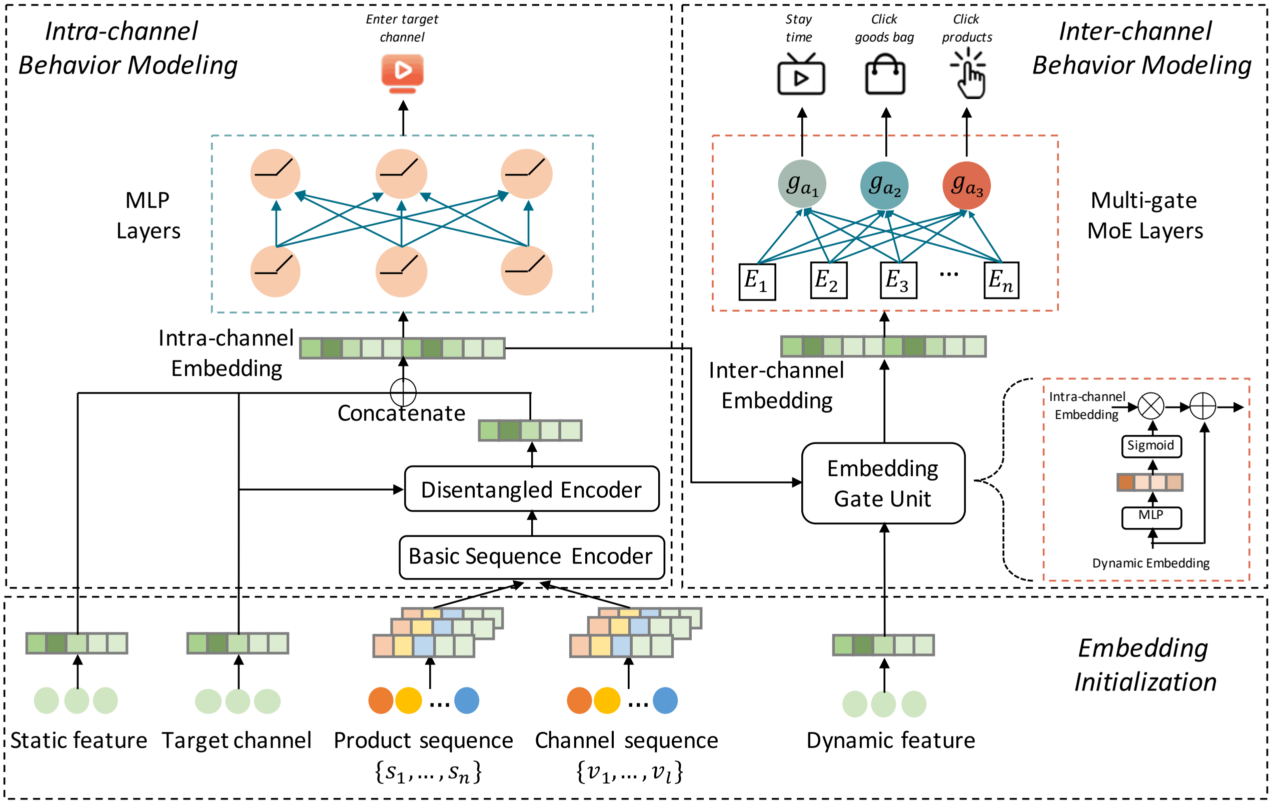

Fig. 3: The overall framework of the proposed eLiveRec model, which formed by cascading Embedding Initialization, Intra-channel Behavior Modeling, and Inter-channel Behavior Modeling. In addition, an Adaptive Multi-task Learning Framework is used to jointly optimize intra- and inter-channel behaviors.

scene and so on; *clicked product* contains *goods category id*, *brand id* and so on; *watched channel* contains *channel id*, *top goods category* and so on; *target channel* contains *channel id*, *channel anchor* and so on; *target channel real-time context* contains *real-time products in the channel*, *the revenue during window time* and so on. It is worthy to note that, we take *user profile* and *spatiotemporal context* as *static context*, while *target channel real-time context* as *dynamic context*.

The raw features are usually represented as high-dimensional sparse vectors by combining multiple one-hot characteristics. We use the common operation **Embedding** transforms the sparse feature into dense valued vector for facilitating follow-up calculations. Then, the embedding matrix of *clicked products sequence* can be represented by $\mathbf{E}_S = [e_{s_1}, e_{s_2}, \cdots, e_{s_n}] \in \mathbb{R}^{n\times d}$, the embedding matrix of *watched channel sequence* can be represented by $\mathbf{E}_V = [e_{v_1}, e_{v_2}, \cdots, e_{v_l}] \in \mathbb{R}^{l\times d}$ similarity. Besides, the embedding vector $\mathbf{e}_{v_t}$ denotes *target channel*, $\mathbf{e}_c$ denotes *static context*, $\mathbf{e}_d$ denotes *dynamic context* for the sake of formulation.

B. Intra-channel Behavior Modeling

The intra-channel behavior modeling module consists of basic sequence encoder, disentangled encoder, and intra-channel behavior prediction components.

1) Basic Sequence Encoder: The existing works has made plenty of attempts in sequence modeling. The sequence encoder can be implemented by many models like RNNs [32], CNNs [33], GNNs [34], [35] and self-attention networks [13], [36]. We also employ traditional sequential model to encode user's sequences from both product and live domain. Specifically, we choose stacking the Transformer blocks as the backbone to model the user's interactive sequences. Given the item representation $\mathbf{H}^{\ell-1}$ at the $(\ell-1)$-th layer, the output of Transformer encoder at the ℓ-th layer is as follows,

$$\begin{aligned}\mathbf{H}^{\ell} &= \mathit{FFN}\big(\mathit{Concat}(head_1, \cdots, head_h)\mathbf{W}^h\big),\\ head_i &= \mathit{Attention}\big(\mathbf{H}^{\ell-1}\mathbf{W}_i^Q, \mathbf{H}^{\ell-1}\mathbf{W}_i^K, \mathbf{H}^{\ell-1}\mathbf{W}_i^V\big),\end{aligned} \tag{3}$$

where $FFN(\cdot)$ represents feed-forward network, h represents the number of heads, the projection matrices $\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V \in \mathbb{R}^{d\times d/h}$, $\mathbf{W}^h \in \mathbb{R}^{d\times d}$. Here, we omit the Residual Network and layer normalization, which used to avoid overfitting, for convenience. The attention mechanism is calculated as follows,

$$\mathit{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathit{softmax}\Big(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\Big)\mathbf{V}, \tag{4}$$

where $\mathbf{Q}$ represents the queries, $\mathbf{K}$ represents the keys, and $\mathbf{V}$ represents the values. The factor $\sqrt{d}$ plays a regulatory role.

Specifically, we use the embedding matrix $\mathbf{E}_S$ and $\mathbf{E}_V$ as the initial state respectively. Based on several Transformer blocks, we can obtain $\mathbf{H}_S = [h_{s_1}, h_{s_2}, \cdots, h_{s_n}] \in \mathbb{R}^{n\times d}$ and

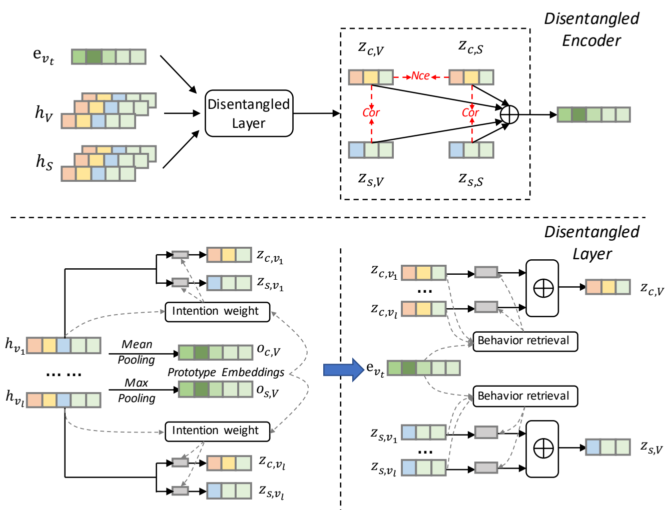

Fig. 4: The architecture of the disentangled encoder, taking live domain as example.

$\mathbf{H}_V = [h_{v_1}, h_{v_2}, \cdots, h_{v_l}] \in \mathbb{R}^{l \times d}$ respectively, which omit the superscript ℓ for brevity.

2) Disentangled Encoder: Intuitively, we model the user's sequential representations from both product and live domains respectively. However, due to the gap between different domains, user may show consistent intentions and domain-specific intentions in different domains occasionally, which illustration in Fig. 2. To deal with the issue, we disentangle the sequential representations to yield consistent intentions and domain-specific intentions respectively. We describe the module by the example of consistent intentions of live domain. The structure of disentangled encoder is shown in Fig. 4.

Intention Weighting. In order to model the user's cross-domain intentions, we should map user's behaviors into consistent intentions and domain-specific intentions respectively. We firstly create two prototypes embeddings $o_{c,V}, o_{s,V} \in \mathbb{R}^d$ by mean pooling and max pooling on the $\mathbf{E}_V$ to denote consistency and specificity respectively. In practice, the pooling prototypes is superior to [25], [26], which using learnable parameters follow a Gaussian distribution as prototypes. Then, we assign weights to each behavior in order to differentiate between two components, namely the consistent and the specific components, based on their distance from two intention prototypes.

$$p(z_{c,v_i}|v_i) = \frac{exp(h_{v_i} \odot o_{c,V})}{\sum_{o \in o_{c,V}, o_{s,V}} exp(h_{v_i} \odot o)}, \tag{5}$$

where $\odot$ denotes the element-wise product, z_{c,v_i} denotes the consistent component of channel v_i, the attention weight $p(z_{c,v_i}|v_i)$ measures how likely the channel v_i is related with the consistent intention.

User Behaviors Retrieval. According to [14], the large amount of noise in user's histories causes the failure of sequential modeling. So the most relevant and appropriate user behaviors should be retrieved from the user's historical sequence. We now introduce intention-related retrieval weights to measure how important of the position i-th channel for modeling the user's consistent/specific intentions. Target attention [10], [14], [37] is widely applied on CTR prediction tasks. Similar to the eq.4, target channel acts as query (Q) and consistent intention-related component z_{c,v_i} acts as key (K) and value (V). The probability $p(c, v_i)$ is obtained by activation of *softmax* function,

$$p(c, v_i) = \frac{exp(e_{v_t} \cdot z_{c,v_i}{}^T)}{\sum_{v_{i'} \in V} exp(e_{v_t} \cdot z_{c,v_{i'}}{}^T)}. \tag{6}$$

Specifically, the user's consistent intentions of live domain can be computed as follows,

$$z_{c,V} = \sum_{v_i \in V} p(c, v_i) \cdot p(z_{c,v_i}|v_i) \cdot h_{v_i}, \tag{7}$$

where $z_{c,V}$ denotes the user's live domain consistently intention representation, $p(z_{c,v_i}|v_i)$ measures how likely the i-th historical channel related with the consistently intention representation, which is described above. Therefore, we can aggregate the intentions collected at the sequences according to $p(z_{c,v_i}|v_i)$ and the important score $p(c, v_i)$. Similarly, $z_{s,V}, z_{c,S}, z_{s,S}$ are also obtained by the same process.

Self-supervised Regularization. In order to ensure the stability of the model in disentangled learning, some regularization [25], [38] are proposed to encourage the disentangled parts to preserve sufficiently different information, to avoid information redundancy. Although we use refined prototype embeddings in this module which are naturally different, there might still be redundancy among the representations. To further encourage the independence among them, we adopt the distance correlation [38] as a regularization. The distance correlation can ensure any two paired embeddings independent. Formally, we define it as follows:

$$\mathcal{L}_{cor} = \sum_{K \in \{S,V\}} \frac{dCov(z_{c,K}, z_{s,K})}{\sqrt{dVar(z_{c,K}) \cdot dVar(z_{s,K})}}, \tag{8}$$

where $dCov(\cdot)$ is the distance covariance between two matrices, and $dVar(\cdot)$ represents its own distance covariance.

Apart from constraining disentangled learning, we also need to ensure that the consistent representations from both two domains are as similar as possible in the latent space. In other words, we maximize the mutual information [39] between $z_{c,V}$ and $z_{c,S}$. According to [40], minimizing the InfoNCE loss [41] is equivalence to maximizing the lower bound of the corresponding mutual information. So we adopt InfoNCE to estimate the mutual information. Formally, for the mutual information of consistent representations, we consider the disentangled intention representations $z_{c,V}$ and $z_{c,S}$ of the same user as the positive pairs, while representations from different user as the negative:

$$\mathcal{L}_{nce} = -log \frac{exp(cos(z_{c,V}, z_{c,S})/\tau)}{\sum_{z_n \sim P_n(z)} exp(cos(z_{c,V}, z_n)/\tau)}, \tag{9}$$

where P_n denote the negative sampling distribution, $cos(\cdot,\cdot)$ is the cosine similarity function, and τ is the temperature that is empirically set to 0.5. In practice, there are some alternative strategies to define the similarity, *e.g.*, cosine distance, euclidean distance. We observe that InfoNCE obtains better performance comparing with cosine/euclidean distance in our practices.

The final self-supervised regularization is as follows,

$$\mathcal{L}_{reg} = \lambda_1 \mathcal{L}_{cor} + \lambda_2 \mathcal{L}_{nce}, \tag{10}$$

where λ_1 and λ_2 are hyper-parameters to control the strengths of self-supervised regularization.

3) Intra-channel Behavior Prediction: Intra-channel behavior prediction is a typical CTR prediction task. To predict whether a user click the target channel v_t, we model it as a binary classification problem. By concatenating the embeddings of static context e_c and the output of the disentangled encoder $(z_{c,V}, z_{s,V}, z_{c,S}, z_{s,S})$ applying to the target channel e_{v_t}, which is called **Intra-channel Embedding**. Besides, we use three MLP layers to further learn the interactions among the dense features, which is standard MLP tower in industrial RS practice [10], [11]. Then we use the *sigmoid* function as the activation unit and adopt cross-entropy loss for the binary classification task.

$$\mathcal{L}_{rec} = -y log(p(v_t|S,V)) - (1-y) log(1-p(v_t|S,V)), \tag{11}$$

where $y \in \{0,1\}$ is the ground-truth label indicating whether the user watches the channel or not, $p(v_t|S,V)$ representing the predicted probability of the target channel being watched.

C. Inter-channel Behavior Modeling

The inter-channel behavior modeling module consists of embedding gate unit, and inter-channel behavior prediction components.

1) Embedding Gate Unit: Historical behaviors record describe the long-term preferences of the user, while real-time preferences are described by the dynamic context of the target channel (*i.e.*, products, live streaming efficiency, anchor) that the user sees after entering the channel. Furthermore, the user's inter-channel behaviors are influenced by both long-term and real-time preferences simultaneously. Inspired by the gated linear unit (GLU) [42], we adopt an embedding gate unit to select what long-term preferences are relevant to real-time preferences, which are utilized to improve the prediction task.

As for the concatenated representation **Intra-channel Embedding**, we can regard it as a feature set $\mathcal{F} = (e_c, e_{v_t}, z_{c,V}, z_{s,V}, z_{c,S}, z_{s,S})$, where $\mathcal{F}_i$ denotes i-th feature in the set $\mathcal{F}$. We use a two-layers MLP network to learn the importance scores of different long-term features in set $\mathcal{F}$, which takes signals related to dynamic context as inputs and outputs weights,

$$w_i = \sigma(MLP(e_d, \Theta_{\mathcal{F}_i})), \tag{12}$$

where $\Theta_{\mathcal{F}_i}$ denotes the parameters related to $\mathcal{F}_i$, the MLP network would generate a weight w_i through *sigmoid* activation σ, the redefined representation was denoted as $\mathcal{F}_i \cdot w_i$. By concatenating the redefined long-term preferences embeddings and real-time preferences embeddings e_d, we can obtain **Inter-channel Embedding** and regard it as the shared input for Multi-gate MoE [29] module.

2) Inter-channel Behavior Prediction: We choose three important indicators, stay time, click goods bag and click products to construct the inter-channel behavior prediction tasks. Among them, *stay time* is considered as a regression task, and the other two behaviors are considered as binary classification tasks. Given y_{a_i} as the ground truth for behavior $a_i, i \in \{$*stay time*, *click goods bag*, *click products*$\}$, the prediction of behavior a_i is represented as $\hat{y}_{a_i}$ (*i.e.*, $p(a_i|S,V,v_t)$) yielded through Multi-gate MoE layers. As for classification tasks, additional activation function $sigmoid$ needs to be introduced. Moreover, we use cross-entropy loss and mean-squared loss to calculate corresponding loss $\mathcal{L}_{task_i}$ respectively.

$$\mathcal{L}_{task_i} = \begin{cases} -y_{a_i} log(\hat{y_{a_i}}) - (1-y_{a_i}) log(1-\hat{y_{a_i}}) & ,C \\ (y_{a_i} - \hat{y_{a_i}})^2 & ,R \end{cases} \tag{13}$$

where C, R represent classification and regression, respectively. As a LSRS of E-commerce, we directly adopt the Multi-gate MoE layers as the shared bottom network, which is widely adopted in both academia and industry like Google [29] and Tencent [43]. A separate gating network is used to selectively utilize different combinations of experts for a specific auxiliary task.

D. Adaptive Multi-task Learning Framework

In this work, our primary task is to estimate the probability that a viewer would like to watch a specific channel (*i.e.*, intra-channel behavior prediction), while the inter-channel behaviors prediction tasks are considered as auxiliary tasks. And the role of the other tasks is to assist in generalization of the primary task. The naive approach [44] to combining multiple auxiliary tasks would be to simply perform a weighted linear sum of the losses for each individual task. However, the process of exploring the weights of auxiliary tasks seems to be too heuristic and irregular, which may cause some problems. Firstly, as the number of auxiliary tasks increases, computational consumption becomes very expensive. Secondly, using fixed weights may limit or even hurt the performance.

We proposed a weighting network to learn weights for different auxiliary tasks thought quantifying the consistency of each auxiliary task to the primary task. Specifically, we quantify the consistency by measuring the cosine similarity of gradients between the auxiliary task and the primary task. The weighting network is a two-layers MLP network, which takes signals of current task's data as input and outputs the weight. We define the weighting network as follows,

$$w_{task_i} = MLP((s(\nabla_\theta \mathcal{L}_{rec}, \nabla_\theta \mathcal{L}_{task_i}), \mathcal{L}_{task_i}, type_i), \Theta_{task_i}), \tag{14}$$

where $s(\cdot,\cdot)$ represents cosine similarity function, $\nabla_\theta \mathcal{L}_{rec}, \nabla_\theta \mathcal{L}_{task_i}$ are the gradients vector of the primary task and i-th auxiliary task loss with respect to the shared parameters respectively, $type_i$ is the type embedding of i-th auxiliary task. Besides, Θ_{task_i} denotes the parameters related with i-th auxiliary task.

Overall, the joint loss can be formulated as follows,

$$\mathcal{L} = \mathcal{L}_{rec} + \sum_{task_i} w_{task_i} \mathcal{L}_{task_i} + \mathcal{L}_{reg}, \tag{15}$$

where w_{task_i} is the output of the weighting network. The regularization term $\mathcal{L}_{reg}$ of disentangled learning are not involved in multi-task learning module. The problem in Eq. 15 is solved by a gradient descent algorithm. We also consider the iterative optimization manner, we leverage the regularization term $\mathcal{L}_{reg}$ to optimize the encoder module. After that, we update all parameters by $\mathcal{L}_{rec} + \sum_{task_i} w_{task_i}$. However, the performance is not satisfactory.

V. Experiments Settings

In this section, we describe dataset used in our experiments, baseline methods, evaluation metrics and implementation details.

A. Dataset

We use a real-world dataset collected from the Taobao Live platform of Alibaba Group[5]. It contains large-scale online live streaming behaviors of users, which cover several consecutive days of users' interactive records on live streaming from the Taobao mobile App's log during September 2022. The dataset is constructed by sampling from the impression and click logs of the online traffic. Each record consists of a record ID, a user ID, a channel ID (exposure to user), a timestamp, an interaction flag (watched the channel or not), and two sequences of the user's watched channels and clicked products, respectively. Among them, we use the staying time to filter the channels and delete the records with staying time below the threshold. In addition, we truncate the sequence lengths into 50. The samples from the consecutive periods are used to construct the training set, and the samples from time immediately following the training set are used as the testing set. The statistics of the dataset are shown in Table I.

B. Baseline Methods

We compare the proposed eLiveRec method with two categories of competitive baselines: CTR models and cross-domain models.

CTR Models.

- WDL+ [5]: This method is a classical industrial RS model, which is composed of wide model and deep model. Besides, we create WDL+ by incorporating sequential information into the base model.

[5]https://www.alibabagroup.com/

TABLE I: Statistics of the experimental dataset.

Index	Dataset	
	Traning	Testing
#Unique feed	1.40×10^8	2.45×10^7
Impression	1.08×10^9	1.92×10^8
Watched	1.27×10^8	2.27×10^7
Avg. impression / feed	7.71	7.86
Avg. watched / feed	0.91	0.93
Avg. product sequence length	47.47	47.44
Avg. channel sequence length	41.29	40.66

- DIN[6] [10]: This method uses the mechanism of attention to capture the similarities between the target item and the previous clicked item.
- DIEN[7] [11]: This method uses two-layer RNNs with attention mechanism to capture evolving user interests.
- HGN[8] [45]: This method uses a hierarchical gating network for sequential recommendation, which works well with its simple gated linear unit (GLU) structure.
- BST [13]: This method leverages the Transformer structure with time information for e-commerce product recommendation.
- CAN[9] [12]: This method leverages co-action unit to model the feature interaction between user behaviors and recommended item.
- ETA [14]: This method proposes a end-to-end target attention to retrieval user behaviors. Besides, we create ETA without long-term interest extraction unit due to online inference real-time constraints.

Cross-Domain Models.

- MiNet[10] [21]: This method jointly models three types of user interests from different domains and utilizes two levels of attentions to fuse these interests.
- DASL[11] [22]: This method proposes dual embedding to represent the cross-domain user and dual attention to model the cross-domain sequential pattern.
- CoNet+[12] [19]: This method leverages the cross connection units to enable dual knowledge transfer across domains between base networks. Besides, we create CoNet+ by incorporating sequential information into the model as user's general feature.
- DTCDR+[12] [46]: This method designs an adaptable embedding-sharing strategy to combine and share the embeddings of common users across domains. Besides, we create DTCDR+ by incorporating sequential information into the model as user's content.
- DDTCDR+[13] [20]: This method develop a novel latent orthogonal mapping to extract user preferences over

[6]https://github.com/zhougr1993/DeepInterestNetwork
[7]https://github.com/mouna99/dien
[8]https://github.com/allenjack/HGN
[9]https://github.com/CAN-Paper/Co-Action-Network
[10]https://github.com/oywtece/minet
[11]https://github.com/lpworld/DASL
[12]https://github.com/RUCAIBox/RecBole-CDR
[13]https://github.com/lpworld/DDTCDR

multiple domains, Besides, we create DDTCDR+ by incorporating sequential information into the model as user's general feature.

C. Evaluation Metrics

We adopt four commonly used performance metrics for evaluation, *i.e.*, AUC, gAUC [10], Precision@K (Prec@K) and Normalized Discounted Cumulative Gain@K (NDCG@K). We calculate Prec@K and NDCG@K according to the record of watched or not. According to Table I, the average impression/watched for a feed is about 7.7 and 0.9 respectively, we experientially set K to 10. The larger value of above metrics indicates better performances.

D. Implementation Details

All the evaluation methods are implemented with TensorFlow[14] 1.4 in Python 2.7, and is conducted on Alibaba's distributed cloud platform. For the offline model training, we conduct distributed training through PS-Worker architecture using CPU, where the number of PS and Worker are 40 and 400, respectively. The model is optimized by distributed training through parameter slicing, etc., where each ps/worker uses 10GB memory. As for online inferring, a single GPU worker with only 32GB memory is used through optimizations such as fp16. For our model, the number of dimensions d is set to 128, set the number of heads in all multi-head attention blocks to 8, the batch size is set to 2048, the initial learning rate is set to 0.05 and we use Adagrad with accumulator decay for optimization. The hyper-parameters for the two self-supervised regularization λ_1 and λ_2 are chosen from $\{0.001, 0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 1.0\}$. In the implementation of vanilla MTL framework, set the weight of the three auxiliary tasks to be 0.3, 0.3 and 0.1 respectively. For baseline methods, we implement each method as following the original papers and report performances under its optimal settings.

VI. Expriments Results

In this section, in order to study the validity of the proposed eLiveRec, we conduct experiments to answer the following research questions:

- **RQ1:** Can the proposed eLiveRec achieve the best recommendation performance compared with baselines?
- **RQ2:** What are the effects of different model components and experimentally settings?
- **RQ3:** How does the proposed eLiveRec perform on the live streaming online platform?
- **RQ4:** What insight can the case study provide?

A. Overall Performance (RQ1)

We implement a variant model **eLiveRec**$_{wo.M}$ of our proposed model eLiveRec, which denotes without adaptive multi-task learning. Based on the CTR baseline methods we chose to compare, we further implemented the corresponding dual sequence variant, *i.e.*, **Dual CTR Model** group. Specifically, we model the user's channel sequence and product sequence in

[14]https://www.tensorflow.org/

TABLE II: The recommendation performance achieved by different methods. The best results are in **boldface**, and the second best results are <u>underlined</u>.

Group	Model	AUC	gAUC	Prec@10	NDCG@10
Single CTR Model	WDL+	0.8224	0.7246	0.4261	0.3795
	DIN	0.8176	0.7194	0.4254	0.3782
	DIEN	0.8206	0.7226	0.4260	0.3780
	HGN	0.8236	0.7271	0.4268	0.3798
	BST	0.8215	0.7254	0.4264	0.3811
	CAN	0.8206	0.7230	0.4264	0.3779
	ETA	0.8304	0.7383	0.4291	0.3854
Dual CTR Model	WDL+	0.8248	0.7298	0.4270	0.3814
	DIN	0.8199	0.7233	0.4261	0.3790
	DIEN	0.8231	0.7268	0.4269	0.3800
	HGN	0.8251	0.7302	0.4273	0.3808
	BST	0.8310	0.7411	0.4289	0.3862
	CAN	0.8235	0.7270	0.4268	0.3796
	ETA	0.8338	0.7451	0.4304	0.3873
Cross-domain Model	MiNet	0.8278	0.7359	0.4279	0.3843
	DASL	0.8236	0.7267	0.4266	0.3801
	CoNet+	0.8335	0.7460	0.4301	0.3879
	DTCDR+	0.8354	0.7499	0.4309	<u>0.3893</u>
	DDTCDR+	0.8319	0.7420	0.4297	0.3870
	eLiveRec$_{wo.M}$	<u>0.8368</u>	<u>0.7506</u>	<u>0.4314</u>	0.3891
	eLiveRec	**0.8398**	**0.7562**	**0.4327**	**0.3911**

parallel, and combine the user preferences learned separately for recommendation. It aims to capture user's preferences in both two domains and optimize live recommendation task with the product domain as a supplement.

Table II summarizes the performance comparison results. Overall, the proposed model outperforms all baseline methods on the industry dataset, in terms of all evaluation metrics. Moreover, we also have the following observations.

Firstly, we verify the necessity of modeling dual sequences. From Table II, the Dual CTR Model group methods significantly outperforms the Single CTR Model group methods. It shows that supplementing user intentions in product domain with more behaviors can improve recommendation performance in live domain with less behaviors. Furthermore, it verifies the necessity of cross-domain intention modeling for LSRS of e-commerce.

Secondly, we observe an interesting point that the performance of some cross-domain recommendation models (MiNet and DASL) don't superior to some excellent CTR models with dual sequences. The possible point is that the better backbone models are adopted, BST and EAT are multi-head attention architecture based method. Another line, EAT is significantly better than DIN, which result also confirms the conclusion. In addition, we create variant cross-domain sequential model (*i.e.*, CoNet+, DTCDR+ and DDTCDR+) by incorporating sequential information into the base mode for fair comparison, which achieve competitive performance than MiNet and DASL due to the using of multi-head attention to model user's sequences. Meanwhile, these models also benefit from the cross-domain transferred knowledge with better performance almost than the all models of Dual CTR Model group.

Thirdly, with a simple but effective gated linear structure, HGN's performance is barely satisfactory. Unfortunately, both of DASL and DDTCDR+ don't perform as expected, which

TABLE III: Ablation study of the disentangled encoder.

Variant	AUC	gAUC	Pre@10	NDCG@10
BST_{Dual}	0.8310	0.7411	0.4289	0.3862
ETA_{Dual}	0.8338	0.7451	0.4304	0.3873
$\text{eLiveRec}_{wo.P}$	0.8378	0.7519	0.4323	0.3900
$\text{eLiveRec}_{wo.C}$	0.8375	0.7508	0.4308	0.3893
$\text{eLiveRec}_{wo.N}$	0.8358	0.7488	0.4309	0.3883
$\text{eLiveRec}_{wo.D}$	0.8343	0.7460	0.4302	0.3881
eLiveRec_{Cos}	0.8325	0.7438	0.4294	0.3850
eLiveRec_{Enc}	0.8255	0.7337	0.4278	0.3795
eLiveRec_{SR}	0.8022	0.6933	0.4142	0.3668
eLiveRec	**0.8398**	**0.7562**	**0.4327**	**0.3911**

means that the alternating optimization paradigm of self-supervised objectives maybe not suitable for the industrial scene of e-commerce live streaming. This also suggests that some new insights need to be injected into emerging industrial practices.

Fourthly, the variant model $\text{eLiveRec}_{wo.M}$ is superior to the cross-domain methods and Dual CTR Model group methods. This shows that our model learns more fine-grained user representation through the disentangled encoder and outperforms the baseline methods. Besides, the performance of eLiveRec is better than $\text{eLiveRec}_{wo.M}$, which illustrates the effectiveness of inter-channel behavior modeling with hierarchical structure through adaptive multi-task learning.

B. Ablation Study and Analysis (RQ2)

Moreover, we also perform ablation study to investigate the impacts of different components and different settings of the proposed method.

1) Disentangled Encoder: To study the importance of sequnce disentangled encoder of eLiveRec, we consider the following eLiveRec variants for evaluation: 1) $\textbf{eLiveRec}_{wo.P}$: we use learnable parameters as prototype embeddings, rather than obtained by pooling operation; 2) $\textbf{eLiveRec}_{wo.C}$: we remove the constraint of disentangled learning, *i.e.*, distance correlation; 3) $\textbf{eLiveRec}_{wo.N}$: we remove the constraint of mutual information, *i.e.*, infoNCE; 4) $\textbf{eLiveRec}_{wo.D}$: we remove the whole disentangled encoder module; 5) $\textbf{eLiveRec}_{Cos}$: we use cosine distance as the self-supervised regularization term to instead infoNCE; 6) $\textbf{eLiveRec}_{Enc}$: we use euclidean distance as the self-supervised regularization term to instead infoNCE; 7) $\textbf{eLiveRec}_{SR}$: we also attempt to adopt regularization terms that minimize the mutual information between domain-specific representations $z_{s,V}$ and $z_{s,S}$. We report the performance of the variants in Table III.

Table III summarizes the performance of eLiveRec variants and the competitive model BST, ETA. We can note that eLiveRec outperforms the variant model $\text{eLiveRec}_{wo.D}$. This indicates the disentangled encoder module helps to improve the live recommendation performance. Moreover, eLiveRec outperforms $\text{eLiveRec}_{wo.C}$ and $\text{eLiveRec}_{wo.N}$, these observations demonstrate that self-supervised regularization can help learn better representations. Besides, eLiveRec outperforms the variant $\text{eLiveRec}_{wo.P}$, which shows that prototype embeddings obtained by pooling can distinguish user's intentions more effectively. The corresponding variant model is always inferior when the cosine/euclidean distance is used to replace InfoNCE as the regularization term. When we try to introduce self-supervised regularization on domain-specific intentions, $w.r.t.$, eLiveRec_{SR}, we find that the performance deteriorates significantly. We guess the possible reason is that the specific intentions of two different domains may be divergent distribution in the latent space, and the improper constraints may lead to the reverse relationship between them resulting a neglect of diversity.

TABLE IV: Ablation study of the adaptive multi-task learning.

Model	Variant	AUC	gAUC	Prec@10	NDCG@10
DIN	single task	0.8199	0.7233	0.4261	0.3790
	vanilla MTL	0.8237	0.7279	0.4269	0.3812
	adaptive MTL	0.8235	0.7285	0.4270	0.3810
HGN	single task	0.8251	0.7302	0.4273	0.3808
	vanilla MTL	0.8247	0.7275	0.4267	0.3812
	adaptive MTL	0.8265	0.7323	0.4278	0.3820
BST	single task	0.8310	0.7411	0.4289	0.3862
	vanilla MTL	0.8260	0.7334	0.4270	0.3834
	adaptive MTL	0.8343	0.7460	0.4299	0.3875
ETA	single task	0.8338	0.7451	0.4304	0.3873
	vanilla MTL	0.8368	0.7494	0.4308	0.3883
	adaptive MTL	0.8371	0.7490	0.4313	0.3889
MiNet	single task	0.8278	0.7359	0.4279	0.3843
	vanilla MTL	0.8285	0.7374	0.4283	0.3848
	adaptive MTL	0.8298	0.7399	0.4286	0.3862
eLiveRec	single task	0.8368	0.7506	0.4314	0.3891
	vanilla MTL	0.8383	0.7551	0.4325	0.3904
	adaptive MTL	0.8398	0.7562	0.4327	0.3911

What's more, $\text{eLiveRec}_{wo.M}$ (find in Table II) achieves better results that $\text{eLiveRec}_{wo.D}$, which indicates that disentangled encoder dominates the performance of eLiveRec, and the adaptive multi-task learning module is a complementary part that can help further improve the recommendation performance.

2) Adaptive Multi-task Learning: To study the effectiveness of the adaptive multi-task learning framework. We choose representative models DIN, HGN, BST, ETA and MiNet to compare with our proposed method eLiveRec, and implement two variants *i.e.*, vanilla weighted multi-task learning and adaptive multi-task learning on the basis of each model, denotes **vanilla MTL** and **adaptive MTL**, respectively. Table IV summarizes the results of variant models.

We observe that both vanilla MTL and adaptive MTL almost perform better than single task, this shows that our foothold is correct. The inter-channel behaviors with hierarchical structure is a significant specialty different from the traditional recommendation, which is helpful for the recommendation performance. As we can see, adaptive MTL performs better than vanilla MTL in terms of almost all metrics. The reason might be that the fixed weights are invariable during the whole training stage, leading to the multi-task learning dominated by a particular task and yielding limited improvements on

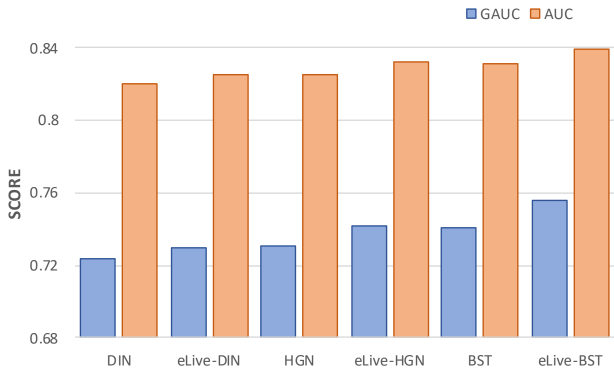

Fig. 5: The performance of DIN, HGN, BST, and eLiveRec with different basic sequence encoders.

primary tasks. Meanwhile, adaptive MTL dynamically adjusts the weights of different tasks by considering the gradient similarity between auxiliary tasks and primary task, which achieve better performance compared with single task.

3) Basic Sequence Encoder: To further investigate the effectiveness of the proposed eLiveRec, we employ other structures to build the basic sequence encoder. Specifically, we consider the following settings of eLiveRec for experiments: 1) **eLive-DIN**: we use the DIN as the backbone structure to build the basic sequence encoder; 2) **eLive-HGN**: we use HGN as the backbone structure to build the basic sequence encoder; 3) **eLive-BST**: The default model that uses the Transformer block as the backbone structure to build the basic sequence encoder.

Fig. 5 shows the performance of eLiveRec with different sequence encoders, as well as the performance of backbone models. We can observe that eLive-DIN, eLive-HGN, and eLive-BST outperform the corresponding backbone encoder models. This indicates that the existing sequential recommendation models can be incorporated into our proposed eLiveRec. Moreover, using Transformer blocks as basic sequence encoder can achiever better performance than other competitive encoders. Through our main contribution, *i.e.*, intention disentangled and adaptive multi-task learning, we can help further improve the recommendation performance on the basis of sequential recommendation.

4) Effect on Different Task: In several recommendation system practices based on multi-task learning, the seesaw phenomenon may exist, *i.e.*, the metrics of a certain task increase, while the metrics of another task decrease. We conduct both online and offline experiments to analyze the phenomenon. Table V shows the gains of the main metrics (*i.e.*, AUC of CTR prediction task) and second-hop metrics, which is the gain of adaptive MTL over vanilla MTL.

We can notice that the performance of the primary task is improved, while the second-hop metrics have a varying degree of increase in addition to the stay time during the offline experiment. In contrast, the results of the online inferences show that an increase of performance in primary task also brings some improvement in second-hop metrics. Since the higher probability of user watching the certain channel indicates that the channel is more in line with user's preferences, which leads to a greater tendency for user to stay in the channel and develop some inter-channel behaviors.

TABLE V: Results from Both Offline and Online A/B Testing.

	AUC	Stay Time	Good Bag	Product	Conversion Rate	Discoverability
offline	+0.26%	-0.21%	+0.39%	+0.17%	N/A	N/A
online	+0.62%	+0.78%	+1.32%	+0.24%	+0.48%	0.18%

5) Effect on Different Sequence Length: Considering the importance of hyper-parameters, we also perform experiments to analyze the impacts of the sequences length. The average channel sequence length in the training set is 41.29 (detailed in Table I), and the value after removing repeated channels is only 27.78. In the live domain, users have a large number of repeated viewing behaviors, which is quite different from the traditional scene. Similar situation can also be found in case study Fig. 8. Considering the strong real-time requirements of live streaming, sequence length 50 is sufficient to cover the dynamic preferences for most users.

In this section, we only study the sequence length of product domain. Fig. 6 shows performance trends with respect to different interactive product sequence length, we can notice that the experimental results improve slightly as the sequence length increases. The performance of the model culminates when the sequence length is 70 and then gradually decreases. However, in our other experiments, we choose the truncation length of 50 to reduce the time complexity of the model, and to keep consistent with the length of the live streaming sequence simultaneously.

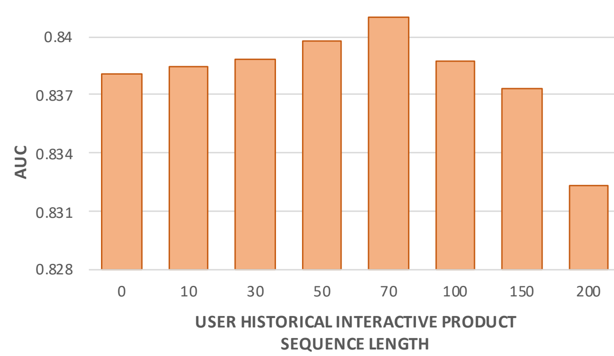

Fig. 6: The performance trends of eLiveRec with respect to different settings of interactive product sequence length.

C. Online Experiments (RQ3)

We conduct an online A/B testing on Taobao Live streaming platform during September 2022, which is under the bucket tests. One bucket is selected for baseline and another bucket for our model. Each bucket serves about 0.5 million users per day.

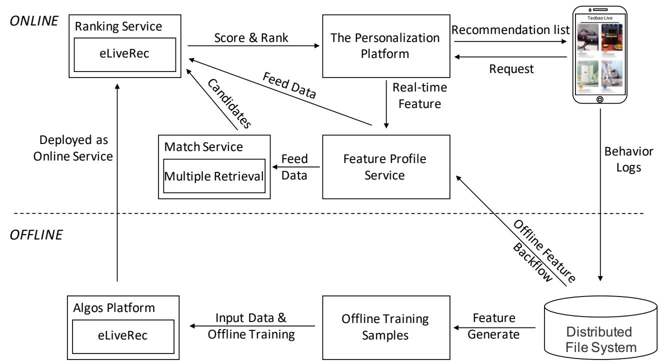

Fig. 7: The deployment pipeline of eLiveRec in Taobao Live Streaming platform.

1) Deployment System and Pipeline: In this section, we introduce the online deployment pipeline of the proposed method. Fig. 7 illustrates the details of deployment system and pipeline. For the workflow of offline is illustrated as follows:

A few behaviors of users' authorized in the Taobao app would be tracked, which collected and stored as log data on distributed file system (DFS) of Alibaba. The log data contain partial information about users' authorized interactive behavior. Based on the log data and DFS of Alibaba, we yield offline training samples by feature generating process. We implemented eLiveRec with TensoFlow and trained the model on the Alibaba's distributed machine learning platform. After that, the model is exported and deployed in the online platform to provide services. The workflow of online is illustrated as follows:

When a user sends an access request on the live streaming homepage, The Personalization Platform (TPP) extracts the user's (*i.e.*, user's real-time interactive behaviors) and channel's (*i.e.*, the statistics during window time, products on sales) real-time feature, which combines offline features produced by DFS backflow. The feature profile service processes the both parts of the features and feeds them to the match service. A candidate set of channels are retrieved by Basic Engine according to several retrieval strategies, *e.g.*, collaborative filtering, embedding similarity. The ranking service ranks the candidates through our deployed eLiveRec and returns the ranked list to TPP. The user can finish the whole request workflow in several milliseconds.

2) Result from Online A/B Testing: We choose ETA with dual sequence as the online baseline. Both baseline and the proposed method are applied in the ranking stage of Taobao Live platform. These methods are loaded with embedding vector representations of features that have been continuously updated for almost two years, then fine-tune the embedding representations using the last one month's data and learn the network parameters simultaneously. After better fitting recent changes in the data distribution, the methods can provide online services to users. During several days testing, eLiveRec contributes up to 1.0% CTR of channels, 5.8% CTR of mounted products and 4.0% conversion rate (page view translates to turnover) promotion compared with the online baseline(*i.e.*, ETA with dual sequence). In terms of stay time, comparable results have been achieved by eLiveRec. Note that in order to reduce the impact on the online business, we choose a strong baseline method, although it may lead to less improvement.

3) Effect of Online Discoverability: Through online experiments, the proposed method shows better performance in discoverability, which means that channels with no exposure for a long time can be exposed by the method. In the 7-, 14-, and 30-days exposure discoverability metrics, the proposed method can achieve more than 1.0% improvement compared with baseline. Especially in the unpopular group of live domain has a relatively higher promotion. Similarly, for low-active users group in the live domain, the stay time, conversion rate and GMV are improved significantly compared with high-active users group. For example, conversion rates increases 28.64 percent in the low-active group and 3.47 percent in the high-active group, and the stay time increases 10.27 percent in the low-active group, while decreased slightly by 1.08 percent in the high-active group. This shows that the proposed method can learn user's fine-grained intentions and accordingly recommend more personalized channels instead of more popular channels. That's because the increased of discoverability exposure means that more unpopular channels that match user's intentions are exposed. Online discoverability has significant value to the ecology of Taobao Live platform, which can improve the experience of users and anchors to facilitate a virtuous cycle of the platform simultaneously.

D. Case Study in Online Platform (RQ4)

A case study is conducted on the Taobao Live platform. For each user, we retrieve her last behavior sequences from both live domain and product domain. Then, eLiveRec and ETA with dual sequence are used to rank the candidates and recommend the top-k to user. Fig. 8 shows two recommendation examples. From the examples, we have the following observations.

- Firstly, users have a large number of repeated viewing behaviors in the live domain, which indicates users may be influenced by the performance style of the live anchor. Since for the same live anchor, the streaming content delivered may be different even for two adjacent days.
- Secondly, in the live domain, users tend to browse some products that need to be displayed, such as clothes. Anchors can better exhibition the upper body effect after wearing them, which is also a major advantage of the e-commerce live streaming scene.
- Thirdly, if the behavior of product domain is not accommodated reasonably in the model, some interactive behaviors with low interest frequency of users in the live stream domain would be covered by the major interest (*e.g.*, autumn clothes in the case). In addition, channels related to crabs and latex pillows also appear in the historical behavior of the live stream domain, but they are not recommended by ETA.

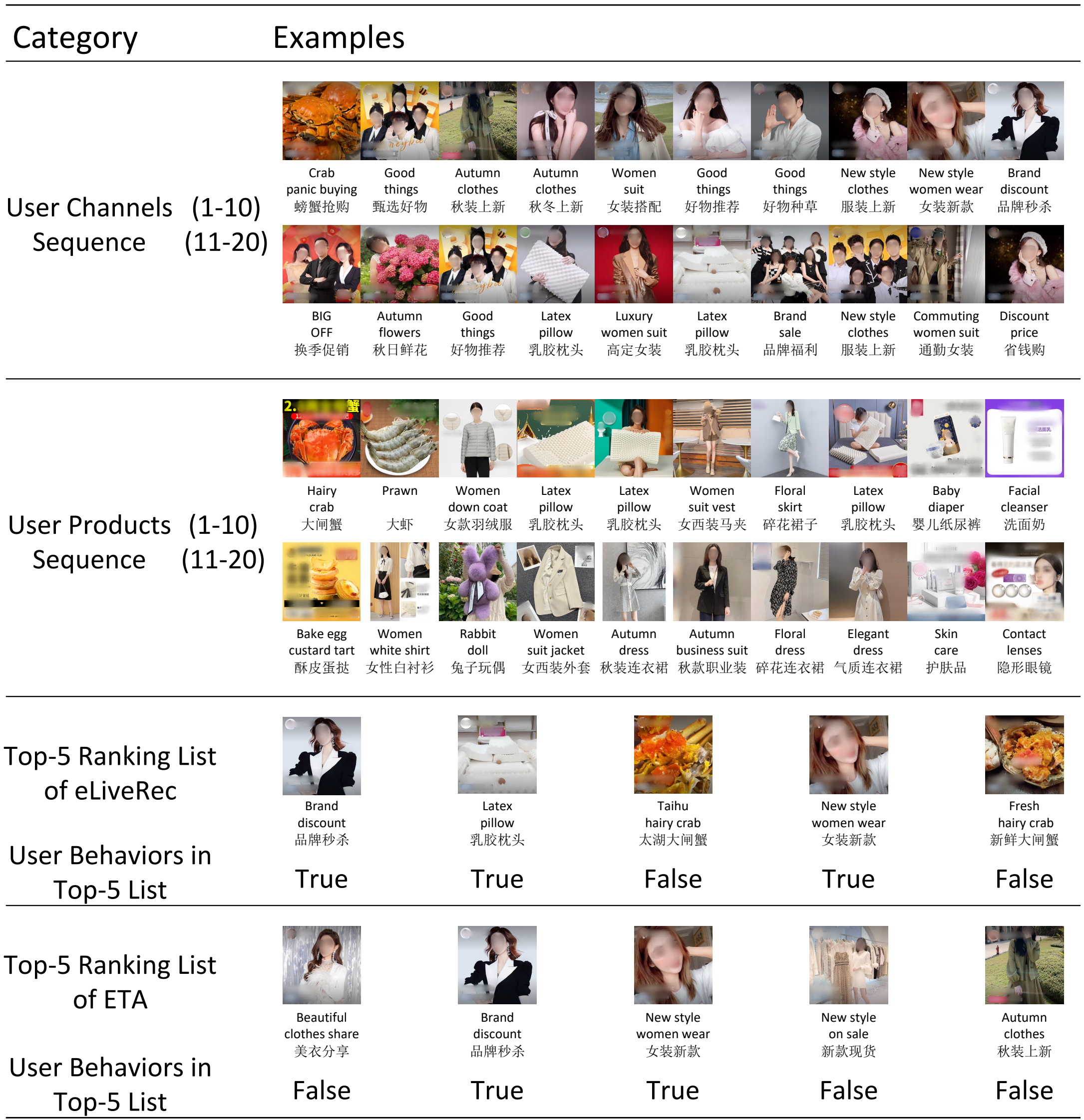

Fig. 8: Examples of Top-5 ranking by eLiveRec and baseline in online platform.

- Fourthly, when modeling dual sequence by disentangled encoder of proposed eLiveRec, consistent intentions are captured and enhanced, such as crab and latex pillow. Besides, the user had interactions with flowers in the live domain, and also had a historical interaction with baby diapers in the product domain, but neither was recommended. This indicates that consistent intentions (from both product and live domains) take precedence over domain-specific intentions. Although the user did not click the crab, she also has a potential intention for the crab based on the analysis of her historical behavior.
- Overall, This case shows the advantages of the proposed eLiveRec compared with competitive model. We can also observe that in real scenes, user's intentions are scattered at times, in which case designing more fine-grained strategies to learn the user's intentions will remain a key challenge of our future work.

VII. Conclusion

In this paper, we study the e-commerce live streaming recommendation task, which is a crucial application of e-commerce platforms. We proposed a novel recommendation method, namely **eLiveRec**, which utilizes user's behaviors in the product domain to assist the live streaming recommendation task. More specifically, eLiveRec utilizes disentangled representation learning to model user's consistent intentions and domain-specific intentions in both product domain and live domain. Moreover, an adaptive multi-task learning framework is designed for the naive hierarchical structure of intra- and inter-channel behaviors, which jointly optimizes user's intra-channel behaviors, as well as her inter-channel behaviors. Extensive experiments on a large-scale industry dataset demonstrate that the proposed eLiveRec method consistently outperforms state-of-the-art baseline recommendation methods.

VIII. Acknowledgement

This research is supported, in part, by the National Key R&D Program of China 2021YFF0900800, NSFC No.62202279, the Shandong Provincial Key Research and Development Program (Major Scientific and Technological Innovation Project) (No.2021CXGC010108), the Shandong Provincial Natural Science Foundation (No.ZR2022QF018), Shandong Provincial Outstanding Youth Science Foundation, the Fundamental Research Funds of Shandong University. This work is also supported, in part, by Alibaba Group through Alibaba Innovative Research (AIR) Program and Alibaba-NTU Singapore Joint Research Institute (JRI), Nanyang Technological University, Singapore.

References

[1] S. Zhang, H. Liu, J. He, S. Han, and X. Du, "Deep sequential model for anchor recommendation on live streaming platforms," *Big Data Min. Anal.*, vol. 4, no. 3, pp. 173–182, 2021. [Online]. Available: https://doi.org/10.26599/BDMA.2021.9020002

[2] J. Rappaz, J. J. McAuley, and K. Aberer, "Recommendation on live-streaming platforms: Dynamic availability and repeat consumption," in *RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021*, H. J. C. Pampín, M. A. Larson, M. C. Willemsen, J. A. Konstan, J. J. McAuley, J. Garcia-Gathright, B. Huurnink, and E. Oldridge, Eds. ACM, 2021, pp. 390–399. [Online]. Available: https://doi.org/10.1145/3460231.3474267

[3] S. Zhang, H. Liu, J. He, S. Han, and X. Du, "A deep bi-directional prediction model for live streaming recommendation," *Inf. Process. Manag.*, vol. 58, no. 2, p. 102453, 2021. [Online]. Available: https://doi.org/10.1016/j.ipm.2020.102453

[4] J. Chen, H. Liu, J. He, and S. Han, "Interaction-aware watching duration prediction on live streaming platforms," in *24th Americas Conference on Information Systems, AMCIS 2018, New Orleans, LA, USA, August 16-18, 2018*. Association for Information Systems, 2018. [Online]. Available: https://aisel.aisnet.org/amcis2018/DataScience/Presentations/10

[5] H. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, R. Anil, Z. Haque, L. Hong, V. Jain, X. Liu, and H. Shah, "Wide & deep learning for recommender systems," in *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, DLRS@RecSys 2016, Boston, MA, USA, September 15, 2016*, A. Karatzoglou, B. Hidasi, D. Tikk, O. S. Shalom, H. Roitman, B. Shapira, and L. Rokach, Eds. ACM, 2016, pp. 7–10. [Online]. Available: https://doi.org/10.1145/2988450.2988454

[6] H. Guo, R. Tang, Y. Ye, Z. Li, and X. He, "Deepfm: A factorization-machine based neural network for CTR prediction," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, C. Sierra, Ed. ijcai.org, 2017, pp. 1725–1731. [Online]. Available: https://doi.org/10.24963/ijcai.2017/239

[7] S. Rendle, "Factorization machines," in *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*, G. I. Webb, B. Liu, C. Zhang, D. Gunopulos, and X. Wu, Eds. IEEE Computer Society, 2010, pp. 995–1000. [Online]. Available: https://doi.org/10.1109/ICDM.2010.127

[8] R. Wang, B. Fu, G. Fu, and M. Wang, "Deep & cross network for ad click predictions," in *Proceedings of the ADKDD'17, Halifax, NS, Canada, August 13 - 17, 2017*. ACM, 2017, pp. 12:1–12:7. [Online]. Available: https://doi.org/10.1145/3124749.3124754

[9] J. Lian, X. Zhou, F. Zhang, Z. Chen, X. Xie, and G. Sun, "xdeepfm: Combining explicit and implicit feature interactions for recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, Y. Guo and F. Farooq, Eds. ACM, 2018, pp. 1754–1763. [Online]. Available: https://doi.org/10.1145/3219819.3220023

[10] G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, and K. Gai, "Deep interest network for click-through rate prediction," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, Y. Guo and F. Farooq, Eds. ACM, 2018, pp. 1059–1068. [Online]. Available: https://doi.org/10.1145/3219819.3219823

[11] G. Zhou, N. Mou, Y. Fan, Q. Pi, W. Bian, C. Zhou, X. Zhu, and K. Gai, "Deep interest evolution network for click-through rate prediction," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 5941–5948. [Online]. Available: https://doi.org/10.1609/aaai.v33i01.33015941

[12] G. Zhou, W. Bian, K. Wu, L. Ren, Q. Pi, Y. Zhang, C. Xiao, X. Sheng, N. Mou, X. Luo, C. Zhang, X. Qiao, S. Xiang, K. Gai, X. Zhu, and J. Xu, "CAN: revisiting feature co-action for click-through rate prediction," *CoRR*, vol. abs/2011.05625, 2020. [Online]. Available: https://arxiv.org/abs/2011.05625

[13] Q. Chen, H. Zhao, W. Li, P. Huang, and W. Ou, "Behavior sequence transformer for e-commerce recommendation in alibaba," *CoRR*, vol. abs/1905.06874, 2019. [Online]. Available: http://arxiv.org/abs/1905.06874

[14] Q. Chen, C. Pei, S. Lv, C. Li, J. Ge, and W. Ou, "End-to-end user behavior retrieval in click-through rateprediction model," *CoRR*, vol. abs/2108.04468, 2021. [Online]. Available: https://arxiv.org/abs/2108.04468

[15] Y. Feng, F. Lv, W. Shen, M. Wang, F. Sun, Y. Zhu, and K. Yang, "Deep session interest network for click-through rate prediction," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, S. Kraus, Ed. ijcai.org, 2019, pp. 2301–2307. [Online]. Available: https://doi.org/10.24963/ijcai.2019/319

[16] W. Ouyang, X. Zhang, L. Li, H. Zou, X. Xing, Z. Liu, and Y. Du, "Deep spatio-temporal neural networks for click-through rate prediction," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, and G. Karypis, Eds. ACM, 2019, pp. 2078–2086. [Online]. Available: https://doi.org/10.1145/3292500.3330655

[17] F. Zhu, Y. Wang, C. Chen, J. Zhou, L. Li, and G. Liu, "Cross-domain recommendation: Challenges, progress, and prospects," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, Z. Zhou, Ed. ijcai.org, 2021, pp. 4721–4728. [Online]. Available: https://doi.org/10.24963/ijcai.2021/639

[18] J. Lian, F. Zhang, X. Xie, and G. Sun, "Cccfnet: A content-boosted collaborative filtering neural network for cross domain recommender systems," in *Proceedings of the 26th International Conference on World Wide Web Companion, Perth, Australia, April 3-7, 2017*, R. Barrett, R. Cummings, E. Agichtein, and E. Gabrilovich, Eds. ACM, 2017, pp. 817–818. [Online]. Available: https://doi.org/10.1145/3041021.3054207

[19] G. Hu, Y. Zhang, and Q. Yang, "Conet: Collaborative cross networks for cross-domain recommendation," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, and H. Wang, Eds. ACM, 2018, pp. 667–676. [Online]. Available: https://doi.org/10.1145/3269206.3271684

[20] P. Li and A. Tuzhilin, "DDTCDR: deep dual transfer cross domain recommendation," in *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, J. Caverlee, X. B. Hu, M. Lalmas, and W. Wang, Eds. ACM, 2020, pp. 331–339. [Online]. Available: https://doi.org/10.1145/3336191.3371793

[21] W. Ouyang, X. Zhang, L. Zhao, J. Luo, Y. Zhang, H. Zou, Z. Liu, and Y. Du, "Minet: Mixed interest network for cross-domain click-through rate prediction," in *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, M. d'Aquin, S. Dietze, C. Hauff, E. Curry, and P. Cudré-Mauroux, Eds. ACM, 2020, pp. 2669–2676. [Online]. Available: https://doi.org/10.1145/3340531.3412728

[22] P. Li, Z. Jiang, M. Que, Y. Hu, and A. Tuzhilin, "Dual attentive sequential learning for cross-domain click-through rate prediction," in

KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021, F. Zhu, B. C. Ooi, and C. Miao, Eds. ACM, 2021, pp. 3172–3180. [Online]. Available: https://doi.org/10.1145/3447548.3467140

[23] M. Yang, F. Liu, Z. Chen, X. Shen, J. Hao, and J. Wang, "Causalvae: Disentangled representation learning via neural structural causal models," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, 2021, pp. 9593–9602. [Online]. Available: https://doi.org/10.1109/CVPR46437.2021.00947

[24] P. He, X. Liu, J. Gao, and W. Chen, "Deberta: decoding-enhanced bert with disentangled attention," in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: https://openreview.net/forum?id=XPZIaotutsD

[25] J. Ma, C. Zhou, H. Yang, P. Cui, X. Wang, and W. Zhu, "Disentangled self-supervision in sequential recommenders," in *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, R. Gupta, Y. Liu, J. Tang, and B. A. Prakash, Eds. ACM, 2020, pp. 483–491. [Online]. Available: https://doi.org/10.1145/3394486.3403091

[26] J. Ma, C. Zhou, P. Cui, H. Yang, and W. Zhu, "Learning disentangled representations for recommendation," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 5712–5723. [Online]. Available: https://dl.acm.org/doi/10.5555/3454287.3454800

[27] F. Liu, Z. Cheng, H. Chen, A. Liu, L. Nie, and M. S. Kankanhalli, "Disentangled multimodal representation learning for recommendation," *CoRR*, vol. abs/2203.05406, 2022. [Online]. Available: https://doi.org/10.48550/arXiv.2203.05406

[28] S. Vandenhende, S. Georgoulis, W. V. Gansbeke, M. Proesmans, D. Dai, and L. V. Gool, "Multi-task learning for dense prediction tasks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3614–3633, 2022. [Online]. Available: https://doi.org/10.1109/TPAMI.2021.3054719

[29] J. Ma, Z. Zhao, X. Yi, J. Chen, L. Hong, and E. H. Chi, "Modeling task relationships in multi-task learning with multi-gate mixture-of-experts," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, Y. Guo and F. Farooq, Eds. ACM, 2018, pp. 1930–1939. [Online]. Available: https://doi.org/10.1145/3219819.3220007

[30] X. Han, Z. Huang, B. An, and J. Bai, "Adaptive transfer learning on graph neural networks," in *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, F. Zhu, B. C. Ooi, and C. Miao, Eds. ACM, 2021, pp. 565–574. [Online]. Available: https://doi.org/10.1145/3447548.3467450

[31] Y. He, X. Feng, C. Cheng, G. Ji, Y. Guo, and J. Caverlee, "Metabalance: Improving multi-task recommendations via adapting gradient magnitudes of auxiliary tasks," in *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, F. Laforest, R. Troncy, E. Simperl, D. Agarwal, A. Gionis, I. Herman, and L. Médini, Eds. ACM, 2022, pp. 2205–2215. [Online]. Available: https://doi.org/10.1145/3485447.3512093

[32] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2016. [Online]. Available: http://arxiv.org/abs/1511.06939

[33] J. Tang and K. Wang, "Personalized top-n sequential recommendation via convolutional sequence embedding," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, WSDM 2018, Marina Del Rey, CA, USA, February 5-9, 2018*, Y. Chang, C. Zhai, Y. Liu, and Y. Maarek, Eds. ACM, 2018, pp. 565–573. [Online]. Available: https://doi.org/10.1145/3159652.3159656

[34] S. Wu, Y. Tang, Y. Zhu, L. Wang, X. Xie, and T. Tan, "Session-based recommendation with graph neural networks," in *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press, 2019, pp. 346–353. [Online]. Available: https://doi.org/10.1609/aaai.v33i01.3301346

[35] Y. Zhang, Y. Liu, Y. Xu, H. Xiong, C. Lei, W. He, L. Cui, and C. Miao, "Enhancing sequential recommendation with graph contrastive learning," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, L. D. Raedt, Ed. ijcai.org, 2022, pp. 2398–2405. [Online]. Available: https://doi.org/10.24963/ijcai.2022/333

[36] W. Kang and J. J. McAuley, "Self-attentive sequential recommendation," in *IEEE International Conference on Data Mining, ICDM 2018, Singapore, November 17-20, 2018*. IEEE Computer Society, 2018, pp. 197–206. [Online]. Available: https://doi.org/10.1109/ICDM.2018.00035

[37] J. Qin, W. Zhang, X. Wu, J. Jin, Y. Fang, and Y. Yu, "User behavior retrieval for click-through rate prediction," in *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, J. Huang, Y. Chang, X. Cheng, J. Kamps, V. Murdock, J. Wen, and Y. Liu, Eds. ACM, 2020, pp. 2347–2356. [Online]. Available: https://doi.org/10.1145/3397271.3401440

[38] G. J. Székely, M. L. Rizzo, and N. K. Bakirov, "Measuring and testing dependence by correlation of distances," *The Annals of Statistics*, vol. 35, no. 6, pp. 2769 – 2794, 2007. [Online]. Available: https://doi.org/10.1214/009053607000000505

[39] X. Zong, Y. Liu, Y. Xu, Y. Zhang, Z. Shen, Y. Yang, and L. Cui, "SAER: sentiment-opinion alignment explainable recommendation," in *Database Systems for Advanced Applications - 27th International Conference, DASFAA 2022, Virtual Event, April 11-14, 2022, Proceedings, Part II*, ser. Lecture Notes in Computer Science, vol. 13246. Springer, 2022, pp. 315–322. [Online]. Available: https://doi.org/10.1007/978-3-031-00126-0_24

[40] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *CoRR*, vol. abs/1807.03748, 2018. [Online]. Available: http://arxiv.org/abs/1807.03748

[41] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation: A new estimation principle for unnormalized statistical models," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, ser. JMLR Proceedings, Y. W. Teh and D. M. Titterington, Eds., vol. 9. JMLR.org, 2010, pp. 297–304. [Online]. Available: http://proceedings.mlr.press/v9/gutmann10a.html

[42] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 2017, pp. 933–941. [Online]. Available: http://proceedings.mlr.press/v70/dauphin17a.html

[43] H. Tang, J. Liu, M. Zhao, and X. Gong, "Progressive layered extraction (PLE): A novel multi-task learning (MTL) model for personalized recommendations," in *RecSys 2020: Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020*, R. L. T. Santos, L. B. Marinho, E. M. Daly, L. Chen, K. Falk, N. Koenigstein, and E. S. de Moura, Eds. ACM, 2020, pp. 269–278. [Online]. Available: https://doi.org/10.1145/3383313.3412236

[44] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*. Computer Vision Foundation / IEEE Computer Society, 2018, pp. 7482–7491. [Online]. Available: https://doi.org/10.1109/CVPR.2018.00781

[45] C. Ma, P. Kang, and X. Liu, "Hierarchical gating networks for sequential recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, and G. Karypis, Eds. ACM, 2019, pp. 825–833. [Online]. Available: https://doi.org/10.1145/3292500.3330984

[46] F. Zhu, C. Chen, Y. Wang, G. Liu, and X. Zheng, "DTCDR: A framework for dual-target cross-domain recommendation," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, W. Zhu, D. Tao, X. Cheng, P. Cui, E. A. Rundensteiner, D. Carmel, Q. He, and J. X. Yu, Eds. ACM, 2019, pp. 1533–1542. [Online]. Available: https://doi.org/10.1145/3357384.3357992