Learning Disentangled Representations for Ads Ranking

Xuxing Chen[1] , **Yan Xie**[1] , **Jin Fang**[1]
[1]Meta
{xuxing, yanxie, fangjin}@meta.com,

Abstract

Online advertising systems rely on ads recommendation to deliver personalized experiences and high-quality ad rankings. To achieve this, large-scale deep learning models are employed to process user-ad interactions for accurate user behavior predictions. However, existing approaches often struggle to maintain optimal performance due to the inherent complexities of data distribution shifts in online serving environments and the heterogeneity of user preferences. To address these challenges, we propose novel causality-aware group learning algorithms that generate disentangled representations to improve ads ranking performance. Our approach focuses on identifying fine-grained segments and specific defects in existing ads ranking models, and developing targeted model architectural or algorithmic patches to mitigate these limitations. Through extensive experiments, we demonstrate the benefits of our approach, showcasing its potential to enhance ads recommendation in modern-scale advertising systems.

1 Introduction

Machine learning has revolutionized advertising and recommendation systems [He *et al.*, 2017; Naumov *et al.*, 2019; Zhang *et al.*, 2022]. It serves as a powerful tool to process different types of features and produce high-quality representations for downstream tasks such as click-through rate (CTR) [McMahan *et al.*, 2013; Guo *et al.*, 2017] prediction and conversion rate (CVR) prediction [Ma *et al.*, 2018], which serve as important metrics to improve user satisfaction of business providers.

Classical machine learning methods for recommendation include logistic regression [McMahan *et al.*, 2013], tree based models [He *et al.*, 2014], and factorization machines [Rendle, 2010], which directly operate on features and labels from users' data. These methods typically require extensive feature engineering efforts while still failing to scale up with the ever-growing user-ad interactions.

Motivated by modern deep learning techniques, there is a significant amount of works dedicated to improve the ads recommendation and ranking through different ways, such as neural collaborative filtering [He *et al.*, 2017], sequential modeling [Kang and McAuley, 2018], multi-task learning [Ma *et al.*, 2018], etc. Deep neural networks have been proven capable of processing a large number of user-ad interactions, sparse and dense features, and multi-modal inputs in an end-to-end manner.

Despite the improved generalization performance and efficiency, most existing ads recommendation models either lack interpretability [Zhang *et al.*, 2020] or require intensive engineering tricks to handle the distribution shifts and biases [Chen *et al.*, 2023]. Besides, the black-box nature of the deep learning models make it difficult for practitioners to scale up the model without sacrificing certain metrics. For example, one may find that although the overall performance of a model is improved through scaling up the model or dataset size, the performance on specific segments, such as users from a particular region, may be drastically worsen, leading to concerns about model explainability [Zhang *et al.*, 2020].

Motivated by this, we developed algorithms for effective causal structure learning from the heterogeneous embeddings in the large-scale ranking models, which offers more causality-aware representation learning and, as a result, improves ads ranking with measurable impacts. Our main contributions can be summarized as follows.

- **Group Structure Learning.** Clicks and conversions in ads can result from multiple underlying causes, such as user interest, ad popularity, or social influence. Traditional models often compress these diverse causes into a single embedding, which can amplify biases like popularity or selection bias. Through causal representation learning, we aim to disentangle these causal factors, mitigate bias, and improve overall recommendation quality. We learn hidden group structures underlying user-ad embeddings to learn more disentangled, causality-aware representations to improve ads ranking prediction
- **Residual Learning.** By disentangling positive and negative user responses, especially as their feature distributions differ significantly, we model their residuals separately. This reduces the variance in both prediction modes, resulting in measurable improvements in model performance.

The rest of the paper is organized as follows. In Section

2 we discuss related work in ads ranking and recommendation systems. Then we develop and analyze our algorithms in Section 3. We report the experimental results in Section 4. Finally we conclude our work in Section 5.

Notation. We denote by $\|\cdot\|$ the ℓ^2 norm for vectors and Frobenius norm for matrices. $\text{sg}(\cdot)$ denotes the stop gradient operator. We use $\text{CELoss}(\cdot, y)$ to represent the cross entropy loss between the logit and label y, and we use $\text{Sigmoid}(\cdot)$ to represent the sigmoid function.

2 Related Work

Multi-Task Ads Recommendation. Among others, multi-task learning (MTL) framework has been widely deployed for ads retrieval and ranking models in the industry. It aims at solving different tasks in a unified model [McCann *et al.*, 2018]. A deep learning based MTL model typically includes two parts – a shared bottom layers and upper task-specific heads. The shared architecture aims at extracting high-quality representations from the large-scale datasets, while the task-specific architecture is dedicated to classification or regression tasks of interest. Generally speaking, the tasks mainly include main tasks and auxiliary tasks. In recommendation system problems, main tasks, such as click-through rate (CTR) [McMahan *et al.*, 2013; Guo *et al.*, 2017] prediction and conversion rate (CVR) prediction [Ma *et al.*, 2018], provide crucial metrics to model the user behaviors for better ads retrieval and ranking. Auxiliary tasks, such as conditional conversion rate prediction and distillation task [Lee and others, 2013; Hinton *et al.*, 2015], are not directly used for ads recommendation, but are proven useful in improving the model performance on main tasks. They are mainly designed to provide additional guidance for the main model. Throughout the paper, we assume that our backbone model of ads recommendation uses an MTL structure.

Representation Learning for Ads Recommendation. Representation learning [Bengio *et al.*, 2013] plays a crucial role in various domains of modern deep learning, such as computer vision [He *et al.*, 2020], natural language processing [Mikolov *et al.*, 2013], graph learning [Kipf and Welling, 2016], etc. It enables superior and efficient feature learning of data input with different domains or modalities. A high-quality representation can usually benefit the training of downstream tasks. In ads recommendation problems, the modalities of data input can vary from text and audio to images and videos with heterogeneous domains like countries and regions. Hence the modern supervised and self-supervised representation learning techniques can be directly adopted. We take generative recommendation [Geng *et al.*, 2022; Rajput *et al.*, 2023] as an example. Inspired by the next-token prediction paradigm in autoregressive models, generative recommendation encoders items as tokens in a shared vocabulary for downstream tasks such as next-item prediction. Vector quantization plays a key role in the tokenization of continuous-valued data input. Algorithms like Vector-Quantized Variational AutoEncoder (VQ-VAE) [Van Den Oord *et al.*, 2017] or Residual-Quantized Variational AutoEncoder (RQ-VAE) [Lee *et al.*, 2022] typically adopt an encoder-decoder architecture and learnable codebooks to perform (hiearchical) clustering, to extract different levels of information hidden in the data input. The idea of vector quantization was later introduced in recommendation tasks [Hou *et al.*, 2023; Rajput *et al.*, 2023] to produce semantic IDs for downstream embedding table lookup.

3 Methodology

In this section, we introduce our main methods. Despite that the recommendation systems has been studied extensively, most existing deep learning based ads models lack explainability or struggle to handle extremely heterogeneous user behaviors and thus easily incur bias [Abdollahpouri, 2020] and fail to generalize well on unseen data. One common pitfall that most methods have is that the newly introduced techniques only focus on improving the overall performance, neglecting the model performance changes on fine-grained segments. As a result, the metrics averaged over all segments might be improved, but the metrics of certain important segments can be drastically degraded. To resolve this issue, we consider learning segments both implicitly and explicitly.

Recall that an MTL model typically consists of the shared architecture and task-specific heads, and thus we aim at enhancing the training of these two parts separately. We will first introduce how to extract implicit grouping structure hidden in the shared architecture in Section 3.1, and then in Section 3.2 we propose a novel loss function motivated by explicit groups given by the labels in task heads.

3.1 Group Structure Learning

We first consider an self-supervised way of classifying the training data into different implicit groups, so as to enhance the model training on different segments. A natural idea is to conduct hierarchical clustering, such as RQ-VAE, over certain types of data. To begin with, we first briefly discuss basics of RQ-VAE, which serves as a hierarchical grouping process in our group learning framework. The RQ-VAE The encoder-decoder architecture produces a reconstruction loss that aims at reconstructing the input through the encoding and decoding processes.

$$\mathcal{L}_{\text{recon}}(x) = \|x - \hat{x}\|^2 \tag{1}$$

where x denotes the input, and $\hat{x} = \text{Decoder}(\text{Encoder}(x))$. In this way the encoded embedding stores useful information that can be further utilized in a hierarchical clustering process captured by the following commitment loss.

$$\begin{aligned}&\mathcal{L}_{\text{commit}}(x, \mathbf{E}) \\ &= \sum_{i=1}^{m} \left(\left\| r^{(i)} - \text{sg}(e_x^{(i)}) \right\|^2 + \mu \left\| \text{sg}(r^{(i)}) - e_x^{(i)} \right\|^2 \right)\end{aligned} \tag{2}$$

where $\mathbf{E} = \{\mathbf{E}^{(1)}, ..., \mathbf{E}^{(m)}\}$ is a collection of the codebooks with $\mathbf{E}^{(i)} = \{e_1^{(i)}, ..., e_n^{(i)}\}$ being codebook i with n codes, m denotes the number of codebooks, $r^{(i)}$ represents the residual vector sent to codebook i,

$$r^{(i+1)} = r^{(i)} - e_x^{(i)} \tag{3}$$

with $r^{(0)} = \text{Encoder}(x)$ being the encoded embedding of input x, $e_x^{(0)} = 0$, and $e_x^{(i)}$ being the code closest to $r^{(i)}$ in codebook i, i.e.,

$$e_x^{(i)} = \underset{e_j^{(i)} \in \mathbf{E}^{(i)}}{\arg\min} \left\| r^{(i)} - e_j^{(i)} \right\|, \quad \forall 1 \leq i \leq m.$$

$\mu > 0$ is a hyperparameter to balance two terms.

Different from existing works that only uses the semantic id in the tokenization, we further utilize the codebooks during the training process to serve as additional embeddings that represent the learned groups – for each input x, we aggregate the closest codes in each layers, and concatenate them back to the input x to enhance the representation learning.

Our method is flexible, in the sense that the data input x in (1) and (2) can vary from intermediate layers' output to input features. We present in Section 4 the numerical results of different choices of x.

3.2 Residual Learning

In addition to the implicit grouping process, we present an explicit grouping strategy in this section. We use x to represent the output of shared architecture and $y \in \{0, 1\}$ to represent the label of the data, with 0 and 1 being negative and positive label respectively. We begin with a briefly review of classical wisdom in constructing loss functions in task-specific heads.

Note that many tasks in ads recommendation are binary classifications. For a data label pair (x, y), the classification tasks typically construct the logit through a learnable function f_{lr} and obtain $f_{\text{lr}}(x)$ as the logit of the cross entropy loss:

$$\mathcal{L}_{\text{lr}}(x, y) = \text{CELoss}(f_{\text{lr}}(x), y), \tag{4}$$

in which the final model prediction will be given by $\text{Sigmoid}(f_{\text{lr}}(x))$. Moreover, one could leverage pseudo label p_y associated with the data and construct

$$\mathcal{L}_{\text{lr\_pseudo}}(x, p_y) = \text{CELoss}(f_{\text{lr\_pseudo}}(x), p_y), \tag{5}$$

in which p_y may be obtained from a teacher model and in this case $\mathcal{L}_{\text{lr\_pseudo}}$ in (5) serves as an auxiliary distillation task loss to help with the prediction [Lee and others, 2013; Hinton *et al.*, 2015]. Thus a typical main and auxiliary task loss can be written as

$$\mathcal{L}_{\text{lr}}(x, y, p_y) = \mathcal{L}_{\text{lr}}(x, y) + \mathcal{L}_{\text{lr\_pseudo}}(x, p_y). \tag{6}$$

To further utilize the information given by the triplet (x, y, p_y) and enhance the training over different groups of data, we propose the following loss function

$$\begin{aligned} \mathcal{L}_{\text{res}}(x, y, p_y) =& (1 - y) \cdot \text{CELoss}(f_0(x), 0 - p_y) \\ &+ y \cdot \text{CELoss}(f_1(x), 1 - p_y), \end{aligned} \tag{7}$$

where f_0 and f_1 are learnable functions and can be chosen as simple multilayer perceptrons (MLPs), with $f_0(x)$ and $f_1(x)$ being the logits of negative and positive samples. Here we naturally split the dataset into positive and negative samples according to their binary labels, and construct personalized model parameters to produce their logits separately.

Additionally, we note that $0 - p_y$ and $1 - p_y$ represent the residuals between the true label and p_y, which serve as causal factors that indicate how well the pseudo-labels fit the groundtruth, revealing the potentially underperforming or underrepresented segments in the models. The idea of considering the residual has been widely used in various domains of machine learning, such as model architecture design [He *et al.*, 2016] and algorithmic designs [Friedman, 2001]. It enables faster convergence of the deep learning algorithms with the variance reduction effects. We thus propose a residual learning loss as

$$\mathcal{L}_{\text{res\_lr}}(x, y, p_y) = \mathcal{L}_{\text{lr}}(x, y, p_y) + \alpha \mathcal{L}_{\text{res}}(x, y, p_y) \tag{8}$$

where $\alpha > 0$ is a hyperparameter to control the effect of residual learning. By introducing the residual loss defined in (7), we encourage the learned representation x from the shared architecture in the model to fit the residual of positive and negative samples, and thus to improve model performance on the main tasks.

4 Experiments

In this section, we report experimental results of CTR and CVR models on large-scale datasets. Most ads recommendation systems adopt a multi-stage paradigm. For example, the retrieval model often adopts a two-tower structure in the shared architecture to handle users' data and ads' data separately, and the model tries to retrieve relatively small amount of ads from a large number of candidates, while the ranking model typically builds on top of hierarchical layers [Zhang *et al.*, 2022] that tackle complex interactions between sparse and dense features, to rank the selected ads for users based on user-ad interactions.

There are two main tasks of interest in ads ranking models, click-through rate (CTR) prediction and conversion rate (CVR) prediction. We conduct experiments on two multi-task ads ranking models which focus on predicting CTR and CVR respectively. Both of them adopt MTL architecture. We test our proposed methods, group structure learning and residual learning in Section 3, as well as certain combinations of them. The results are summarized in Table 1. We apply our group structure learning technique on top of user feature, user-ad feature, and the output of the shared architecture. Residual learning loss is directly applied on the output of the shared architecture. More details can be found in the captions of Table 1.

Metrics. We use normalized entropy (NE) [He *et al.*, 2014], a metric widely used in recommendation systems and ads ranking, to evaluate the model performance on different tasks such as CTR and CVR prediction. In particular, for the CTR model we consider the model performance on four types of click-through rates prediction – (overall) click, link click, website click, and profile click.

Observations. We can observe from Table 1 that our methods consistently improve the model performance over the existing baseline, across all different types of click-through rate prediction, indicating that our proposed methods improve the model performance on both the overall and fine-grained segments.

Table 1: NE gains of our methods on different tasks as compared to the existing baseline. We use GSL(#) to represent group structure learning applied on $\# \in \{$user feature, user-ad feature, representation$\}$, where 'representation' denotes the output of the shared architecture. We use ResL to denote the residual learning loss applied on the output of the shared architecture.

	Click	Link-Click	Website-Click	Profile-Click
Baseline	0.00%	0.00%	0.00%	0.00%
GSL(user feature)	0.072%	0.078%	0.071%	0.075%
GSL(user feature), ResL	0.077%	0.10%	0.070%	0.097%
GSL(user-ad feature)	0.074%	0.081%	0.067%	0.079%
GSL(representation)	0.097%	0.086%	0.081%	0.036%

5 Conclusions

In this paper, we propose group structure learning and residual learning to better solve multi-task ads recommendation problems, where the former learns implicit groups in a self-supervised way for better representation learning, and the latter constructs an auxiliary task that leverages the explicit groups in a supervised way to boost the performance on main tasks. Through extensive numerical experiments on large-scale datasets, we demonstrate the benefits of our method over existing baselines. We hope our study can provide valuable insights into the next-generation architecture design for multi-task learning models of recommendation system.

Ethical Statement

There are no ethical issues.

References

Himan Abdollahpouri. *Popularity bias in recommendation: A multi-stakeholder perspective*. PhD thesis, University of Colorado at Boulder, 2020.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35:1798–1828, 2013.

Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems*, 41(3):1–39, 2023.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM conference on recommender systems*, pages 299–315, 2022.

Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. Deepfm: a factorization-machine based neural network for ctr prediction. *arXiv preprint arXiv:1703.04247*, 2017.

Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, et al. Practical lessons from predicting clicks on ads at facebook. In *Proceedings of the eighth international workshop on data mining for online advertising*, pages 1–9, 2014.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182, 2017.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF con-*

ference on computer vision and pattern recognition, pages 9729–9738, 2020.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Yupeng Hou, Zhankui He, Julian McAuley, and Wayne Xin Zhao. Learning vector-quantized item representation for transferable sequential recommenders. In *Proceedings of the ACM Web Conference 2023*, pages 1162–1171, 2023.

Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE, 2018.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11523–11532, 2022.

Xiao Ma, Liqin Zhao, Guan Huang, Zhi Wang, Zelin Hu, Xiaoqiang Zhu, and Kun Gai. Entire space multi-task model: An effective approach for estimating post-click conversion rate. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1137–1140, 2018.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*, 2018.

H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1222–1230, 2013.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Maxim Naumov, Dheevatsa Mudigere, Hao-Jun Michael Shi, Jianyu Huang, Narayanan Sundaraman, Jongsoo Park, Xiaodong Wang, Udit Gupta, Carole-Jean Wu, Alisson G Azzolini, et al. Deep learning recommendation model for personalization and recommendation systems. *arXiv preprint arXiv:1906.00091*, 2019.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Tran, Jonah Samost, et al. Recommender systems with generative retrieval. *Advances in Neural Information Processing Systems*, 36:10299–10315, 2023.

Steffen Rendle. Factorization machines. In *2010 IEEE International conference on data mining*, pages 995–1000. IEEE, 2010.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Yongfeng Zhang, Xu Chen, et al. Explainable recommendation: A survey and new perspectives. *Foundations and Trends® in Information Retrieval*, 14:1–101, 2020.

Buyun Zhang, Liang Luo, Xi Liu, Jay Li, Zeliang Chen, Weilin Zhang, Xiaohan Wei, Yuchen Hao, Michael Tsang, Wenjun Wang, et al. Dhen: A deep and hierarchical ensemble network for large-scale click-through rate prediction. *arXiv preprint arXiv:2203.11014*, 2022.