Invariant Representation Learning for Multimedia Recommendation

Xiaoyu Du
Nanjing University of Science and Technology
duxy@njust.edu.cn

Zike Wu
South China University of Technology
zikewu43@gmail.com

Fuli Feng
University of Science and Technology of China
fulifeng93@gmail.com

Xiangnan He
University of Science and Technology of China
xiangnanhe@gmail.com

Jinhui Tang*
Nanjing University of Science and Technology
jinhuitang@njust.edu.cn

ABSTRACT

Multimedia recommendation forms a personalized ranking task with multimedia content representations which are mostly extracted via generic encoders. However, the generic representations introduce spurious correlations — the meaningless correlation from the recommendation perspective. For example, suppose a user bought two dresses on the same model, this co-occurrence would produce a correlation between the model and purchases, but the correlation is spurious from the view of fashion recommendation. Existing work alleviates this issue by customizing preference-aware representations, requiring high-cost analysis and design.

In this paper, we propose an **Inv**ariant **R**epresentation **L**earning Framework (InvRL) to alleviate the impact of the spurious correlations. We utilize environments to reflect the spurious correlations and determine each environment with a set of interactions. We then learn invariant representations — the inherent factors attracting user attention — to make a consistent prediction of user-item interaction across various environments. In this light, InvRL proposes two iteratively executed modules to cluster user-item interactions and learn invariant representations. According to the learned invariant representations, InvRL trains a final recommender model thus mitigating the spurious correlations. We demonstrate InvRL on a cutting-edge recommender model UltraGCN and conduct extensive experiments on three public multimedia recommendation datasets, Movielens, Tiktok, and Kwai. The experimental results validate the rationality and effectiveness of InvRL.

CCS CONCEPTS

• **Information systems → Multimedia information systems**.

KEYWORDS

Multimedia Recommendation, Multimedia Representation Learning, Invariant Learning, Spurious Correlation

ACM Reference Format:
Xiaoyu Du, Zike Wu, Fuli Feng, Xiangnan He, and Jinhui Tang. 2022. Invariant Representation Learning for Multimedia Recommendation. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3503161.3548405

1 INTRODUCTION

Multimedia recommendation has become a core application in various online platforms for e-commerce [6], social media [40], video sharing [38], *etc.* Multimedia recommender models provide personalized services by learning user preferences from both historical interactions and multimedia item contents, including text, images, audio, and videos. The core is encoding the multimedia contents into proper semantic item representations matching user preferences, which is typically achieved by deep neural network based encoders such as ResNet [9] and BERT [5]. These encoders are designed for generic content understanding tasks such as visual classification, objective recognition, and text classification, even pre-trained over datasets of these tasks.

However, the generic content representations focus on general semantic information rather than the factors that attract user preferences, thus introducing spurious correlations [1], *i.e.,* the irrelevant properties from the recommendation perspective. This will hinder the recommender model to capture genuine user preferences and provide accurate recommendations. Figure 1 exhibits two examples with the issue of spurious correlations. EX#1 is an e-commerce example, where a user recently viewed two dresses with clear style and pretty models as shown in the commodity photos. The generic visual encoder will extract both style and model representations, but the recommender can hardly differentiate the true preference over clothes style or model thus make accurate recommendation. EX#2 presents a movie sharing example, where a user watched two superhero movies both with tags of "*Hero*", "*Handsome*", and "*Fly*". Due to including more of these three tags, the recommender may prefer "*Harry Potter*" to "*Wonder Woman*". To alleviate the impact of spurious correlations, it is critical to remove the above preference-independent but strongly behavior-correlated factors from the multimedia representations.

[*] This work is supported by the National Key Research and Development Program of China (2021ZD0111802), the National Natural Science Foundation of China (61925204 and 62172226), and the 2021 Jiangsu Shuangchuang (Mass Innovation and Entrepreneurship) Talent Program (JSSCBS20210200). Jinhui Tang is the corresponding author.

MM '22, October 10–14, 2022, Lisboa, Portugal
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3548405

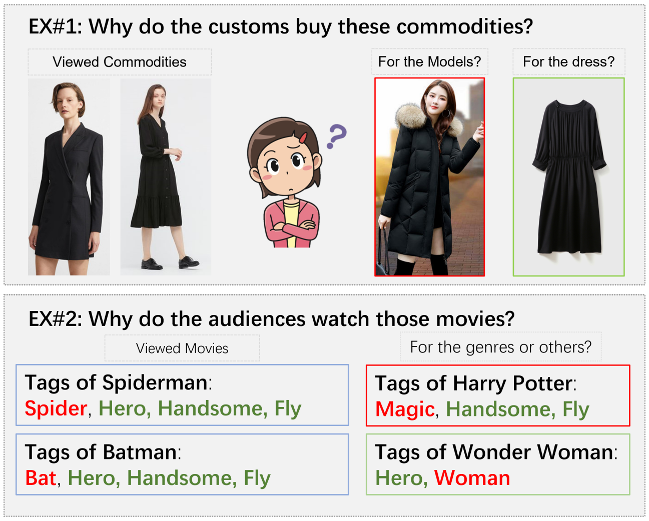

Figure 1: Two examples to illustrate the spurious correlations in multimedia recommendation.

To approach this issue, existing researches mainly customize preference-aware representations [4, 22, 24, 33, 42]. On one hand, the existing methods avoid generic multimedia encoders and extract preference-aware representations with specifically designed multimedia models such as aesthetic-aware models [42] and style-aware models [24] for fashion recommendation, and modal-fusion models [30] for music recommendation. On the other hand, the existing methods adopt generic encoders and focus on blocking the effect of spurious multimedia representations [33]. Nevertheless, the existing methods face the limitation of domain-specific analysis and design, and thus can hardly be generalized. Therefore, we pursue a generic solution to automatically recognize preference-aware representation from the generic multimedia encoders.

Recent advances in invariant risk minimization (IRM) [1] show that the spurious correlations in visual tasks are separable. They illustrate the example of classifying the cows and camels where the cow and the green pasture co-occurrence in many cases. As so, the cow and the green pasture form a spurious correlation: whether an animal is a camel or a cow would rely on whether the background is green. To address this issue, IRM aims to present invariant correlations that do not depend on the environment. Thus it learns invariant data representation by minimizing the max training error across multiple environments. EX#1 presents the spurious correlations between the dresses and the models. Intuitively, we can set two environments w/ and w/o models, and learn invariant item representations across them to obtain accurate preference-aware representations. However, due to the too many unlabeled inherent factors impacting recommendation, to date, there is no study about invariant representation learning in the field of multimedia recommendation. Inspired by Heterogeneous Risk Minimization (HRM) [23], we aim to exploit a manner to partition the heterogeneous environments automatically.

In this paper, we propose an **Inv**ariant **R**epresentation **L**earning framework (InvRL) to alleviate the impact of spurious correlations. Specifically, we divide the raw multimedia representations into two parts: variant and invariant representations, where the variant representations lead to spurious correlations while the invariant representations reflect the real user preferences. We follow the concept of 'environment' in IRM [1] and form it as a subset of user behaviors that can be restored with the variant representations. In this light, we devise two iteratively executed modules to cluster the user-item interactions into heterogeneous environments and learn invariant representations across the environments. By repeating the two modules multiple times, the invariant representations are obtained to promote a final recommender model. To verify the effectiveness of InvRL, we instantiate InvRL over UltraGCN [27] and conduct extensive experiments on three public datasets, Movielens, Tiktok, and Kwai. The experimental results demonstrate that InvRL achieves state-of-the-art multimedia recommendation performance. This reveals the significance of the preference-aware invariant patterns. In addition, the ablation studies verify the existence and separability of the environments.

To summarize, the main contributions are as follows:

(1) We reveal the issue of spurious correlations in multimedia recommendation and address the issue from the perspective of invariant learning, which is new for recommendation.
(2) We propose a new multimedia recommendation framework named InvRL, which learns invariant item representations to alleviate the impact of spurious correlations.
(3) We instantiate the framework over UltraGCN and conduct extensive experiments over three public datasets, verifying the existence and separability of spurious correlations, and validating the rationality and effectiveness of InvRL.

2 METHOD

In this section, we introduce the formulation of multimedia recommendation, the invariant learning perspective of blocking spurious correlations, and the proposed **Inv**ariant **R**epresentation **L**earning framework (InvRL). In this section, we use upper case bold letters to denote matrices, lower case bold letters to denote column vectors, non-bold letters to represent scalars, and calligraphic letters to represent sets. In addition, we adopt $\langle \cdot, \cdot \rangle$ and $\odot$ to indicate the inner product and element-wise product, respectively.

2.1 Problem Formulation

• **Multimedia recommendation.** This task aims to learn a recommender model $\Gamma(u, i, \mathbf{c}_i|\Theta)$ to predict the preference of user u on item i with considerations of multimedia content representations $\mathbf{c}_i$ of the item, which is parameterized by Θ. $\mathbf{c}_i$ is a representation vector that represents the multimedia contents of the item, including text, image, audio, and video. Without loss of generality, we assume $\mathbf{c}_i$ is extracted by generic deep neural network based encoders[1]. A default choice to learn a multimedia recommender model is to optimize the parameters Θ over historical user-item interactions $\mathcal{R} = \{(u, i)|u \in \mathcal{U} \bigwedge i \in \mathcal{I} \bigwedge r_{ui} = 1\}$, where $\mathcal{U}$ and $\mathcal{I}$ to denote the set of N users and M items, r_{ui} is a binary indicator of the interaction status. Formally, the optimization objective is:

$$\arg\min_{\Theta} \mathcal{L}\left(\Gamma(u, i, \mathbf{c}_i|\Theta)|\mathcal{R}^{tr}\right), \tag{1}$$

where $\mathcal{L}(\cdot)$ denotes a recommendation loss such as Binary Cross-Entropy. $\mathcal{R}^{tr} = \mathcal{R} \bigcup \mathcal{R}^{-}$ denotes the training set where $\mathcal{R}^{-}$ includes

[1] $\mathbf{c}_i$ is typically a concatenation of outputs from multiple encoders.

randomly selected negative samples with $r_{ui} = 0$. Note that we omit the regularization term (*e.g.,* l_2-norm) for briefness which is commonly used for preventing overfitting.

• **Blocking spurious correlations.** In an ideal case, all representations in $\mathbf{c}_i$ describe multimedia contents that match user preferences. However, $\mathbf{c}_i$ comes from generic encoders, which not only includes preference-aware representations. Taking fashion recommendation as an example, $\mathbf{c}_i$ includes both representations of clothes (*e.g.,* colour and style) and representations of models (*e.g.,* gender). Such redundant representations will result in spurious correlations (*e.g.,* pretty model to interaction in Figure 1), when learning preferences from historical interactions. To pursue accurate recommendation, we set blocking the spurious correlation as an additional target for multimodel recommender learning.

2.2 Invariant Learning for Recommendation

Invariant learning [1, 23] is an emerging technique for blocking spurious correlations. The key belief is that spurious correlations are unstable across heterogeneous environments, *e.g.,* the correlation between the background green grass and label *cow* is unstable across the images of natural photos and sketches. In this light, invariant learning pushes machine learning models to focus on invariant representation across environments, *e.g.,* optimizing model parameters with an invariant risk minimization objective. Formally,

$$\mathcal{L}_{IRM} = \mathbb{E}_{e\in\mathcal{E}}\mathcal{L}^e + \alpha\|\mathbf{Var}_{e\in\mathcal{E}}(\nabla_\Theta\mathcal{L}^e)\|^2, \tag{2}$$

where the second term is the constraint over the variance across environments. $\mathcal{L}^e$ is an environment specific loss:

$$\mathcal{L}^e = \mathcal{L}\left(\Gamma(u, i, \mathbf{c}_i\})|\Theta)|\mathcal{R}_e^{tr}\right). \tag{3}$$

$\mathcal{R}_e^{tr}$ is the set of training samples under the e-th environment.

There are many natural environments in recommendation tasks, *e.g.,* , regions for the localization-aware recommendation, genres for the movie recommendation, commodity categories for e-commerce recommendations, *etc.* However, these situations rely on specific scenarios thus requiring analysis and designs. Therefore, in this work, we propose the invariant representation learning framework for multimedia recommendation to partition the heterogeneous environments automatically and learn invariant representation across the environments.

2.3 InvRL Framework

Here we present the proposed framework InvRL. We first exhibit the workflow of our framework in Figure 2. There are six key modules (M1-M6) to generate the final model. M1 denotes the content extraction module which is done by pre-trained models. Therefore, InvRL actually starts from M2 which masks part of the content representations to capture the variant representations. According to that, M3 constructs independent interaction environments by dividing the original interactions to several subsets. Subsequently, M4 learns an invariant mask to make stable predictions across the subsets. The opposite of invariant mask is the variant mask, which is fed into the upcoming M2 round. By executing the circle M2-M3-M4 for several times, we obtain stable environments and invariant mask. Then, with M5 and M6, we train a final model on the invariant representations. We will illustrate M2 and M5 in Section 2.3.1, elaborate M3 in Section 2.3.2, present M4 in Section 2.3.3, and demonstrate M6 in Section 2.3.4.

2.3.1 *Invariant and Variant Representations.*

The modules M2 and M5 separate the raw multimedia representations into invariant and variant parts. Toward this target, we devise an invariant mask $\mathbf{m} \in \mathbb{R}^D$ to separate the raw multimedia content representation c_i in to invariant representation Φ_i and Ψ_i, where D is the dimension of the raw content representations. The invariant representation Φ_i is defined as,

$$\Phi_i = \mathbf{m} \odot \mathbf{c}_i. \tag{4}$$

After removing the invariant part, the rest is the variant representation Ψ_i, which is formulated as,

$$\Psi_i = (\mathbf{1} - \mathbf{m}) \odot \mathbf{c}_i. \tag{5}$$

In addition, we adopt $\Phi = \{\Phi_i | i \in \mathcal{I}\}$ and $\Psi = \{\Psi_i | i \in \mathcal{I}\}$ to demonstrate the sets of invariant and variant representations, respectively. Obviously, the key to the invariant item representations lies in the generation of the invariant mask $\mathbf{m}$. It is optimized gradually during the recurrent modules M2-M3-M4. We elaborate on them within two procedures, Environment Partition and Mask Generation.

2.3.2 *Environment Partition.*

Environment Partition procedure relies on the module M3 shown in Figure 2, which takes in the observed user-item historical interactions and outputs an environment set $\mathcal{E}$, where each environment $e \in \mathcal{E}$ reflects a type of spurious correlations between users and items. Taking the environments as references we separate the observed interactions into the environments. Correspondingly, we devise a two-phase partition approach.

• **Environment Learning Phase.** Let $\mathcal{R}_e$ be the set of interactions in the environment e, we now try to express the environment in this phase. Intuitively, the key differences among different environments are their viewpoints on spurious correlations. Thus we model the environment according to the spurious correlations. That is, for the interactions in the environment e, we learn a predictive model $\Gamma^{(e)}$ according to the variant representations Ψ. In another word, to describe the environment e, we learn the predictive model by,

$$\arg\min_{\Theta_e} \mathcal{L}(\Gamma^{(e)}(u, i, \Psi_i|\Theta_e)|\mathcal{R}_e^{tr}), \tag{6}$$

where $\mathbf{\Theta}_e$ indicates the model parameters. Note that the environments are learned independently. They focus on different set of interactions and gain different $\Gamma^{(e)}$. For more details about the backbone model Γ, please refer to Section 2.3.5.

• **Interaction separation Phase.** In this phase, we already have E environments, which indicate E different kinds of spurious correlations. Thus the interactions should be separated according to their spurious correlations. To differentiate the interactions (u, i) in environment, we adopt the following formula,

$$e(u, i) = \arg\max_{e\in\mathcal{E}} \Gamma^{(e)}(u, i, \Psi_i|\Theta_e). \tag{7}$$

That means the interactions belong to the environment with the highest probability to recognize the interaction. Note that the prediction is based on the variant representations Ψ, which expresses spurious information.

Finally, the two phases are run alternatively till converged. Then the interaction partition results $\{\mathcal{R}^{(e)} | e \in \mathcal{E}\}$ are left for the following mask generation procedure.

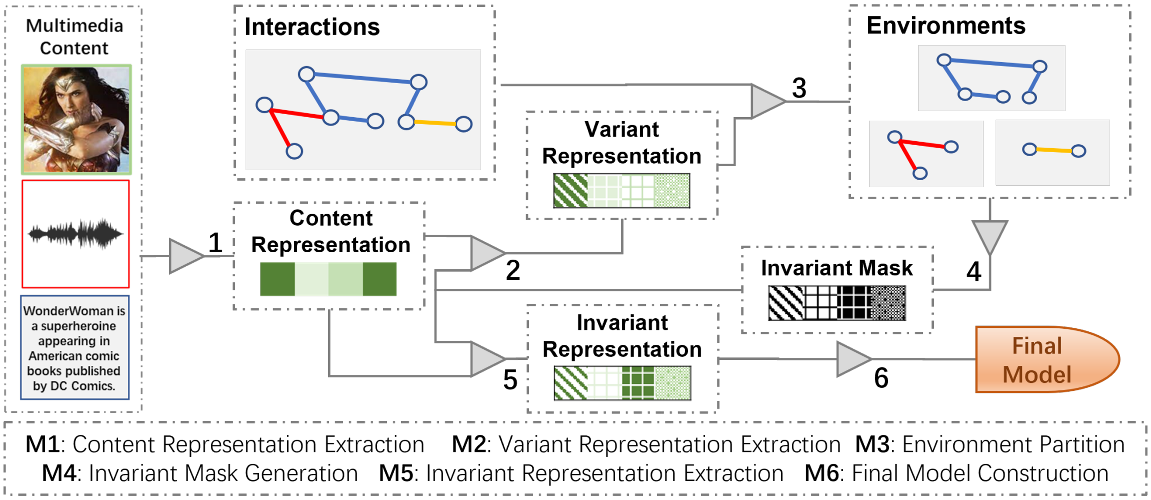
Multimedia Content
WonderWoman is a superheroine appearing in American comic books published by DC Comics.
Interactions
Environments
Variant Representation
Content Representation
Invariant Mask
Invariant Representation
Final Model
1
2
3
4
5
6
M1: Content Representation Extraction M2: Variant Representation Extraction M3: Environment Partition
M4: Invariant Mask Generation M5: Invariant Representation Extraction M6: Final Model Construction

Figure 2: The overall workflow of our framework. The numbered grey triangles indicate the key modules of our approach.

2.3.3 Mask Generation. As shown in Figure 2, the module M4 takes in the environments and generate the invariant mask and variant mask, respectively. Actually, Equation 4 and Equation 5 demonstrate that the two mask relies on a same vector $\mathbf{m} = (m_1, ..., m_D)$. We highlight that, $\mathbf{m}$ is used to generate invariant item representation. Thus we aim to seek an $\mathbf{m}$ that can guarantees the predictive model performs consistently across the environments. We train a cross-environment model and tune the value of $\mathbf{m}$ in two steps. Following the previous work [23], we define $\mu = (\mu_1, ..., \mu_D)$ as,

$$\mu_i = \max\{0, \min\{1, m_i + \epsilon\}\}, \tag{8}$$

where $\epsilon \sim N(0, \sigma^2)$. Then, the predictive function is,

$$\Gamma^{mask}(u, i, \mu \odot \mathbf{c}_i | \Theta^{mask}). \tag{9}$$

- **Initialization.** As $\mathbf{m}$ is for the separation of the raw multimedia representations, we separate representation equally before training. m_i is initialized with the value of 0.5. Additionally, we initialize the predictive model Γ^{mask} with the parameters from a pre-trained raw model of Γ.
- **Optimization.** In order to maintain the consistency of Γ^{mask} across the environments, we utilize the learning target of HRM [23] and improve it to adapt our task. The object function is composed of two major terms,

$$\mathcal{L}_{mask} = \mathbb{E}_{e \in \mathcal{E}} \mathcal{L}^e + \alpha \| \mathbf{Var}_{e \in \mathcal{E}} (\nabla_{\Theta^{mask}} \mathcal{L}^e) \odot \mu \|^2 + \lambda \|\mathbf{m}\|^2, \tag{10}$$

where the first term is the ordinary recommendation loss, the second term is the constraint across environments, and the third term is a regularization term. $\mathcal{L}^e$ is the average loss value inside the environment e, that is,

$$\mathcal{L}^e = \mathcal{L}(\Gamma^{mask}(u, i, \mu \odot \mathbf{c}_i) | \Theta^{mask} | \mathcal{R}_e^{tr}). \tag{11}$$

Accordingly, the mask $\mathbf{m}$ is optimized to minimize $\mathcal{L}_{mask}$. To ensure $0 \leq m_i \leq 1$, after each iteration, we clip the mask $\mathbf{m}$ with,

$$m_i \leftarrow \max\{0, \min\{1, m_i\}\}. \tag{12}$$

When the predictive model Γ^{mask} converged, the invariant and invariant representations can be generated with Equation 4 and Equation 5, respectively.

2.3.4 Final Predictive Model. By repeating the running flow M2-M3-M4 (as shown in Figure 2) for T times till converged, the invariant mask becomes stable. Thus, we learn the final predictive model according to the invariant representation generated by M5. The learning target is a bit different from Equation 1, as

$$\arg\min_{\Theta^*} \mathcal{L}(\Gamma^*(u, i, \Phi_i | \Theta^*) | \mathcal{R}^{tr}). \tag{13}$$

Summarily, the overall training process is presented in Algorithm 1.

Algorithm 1: The overall training process.

```
Data: $\mathcal{R}, \mathcal{R}^-, \mathcal{R}^{tr}$
Result: Final Predictive Model $\Gamma^*(u, i | \Theta^*, \Phi)$
for $i \leftarrow 1$ to $T$ do
    /* M3                                  */
    do
        for $e \in \mathcal{E}$ do
            Optimize $\Gamma^{(e)}$ via Eq. (6);
        end
        for $e \in \mathcal{E}$ do
            Compute $\mathcal{R}_e$ via Eq. (7);
        end
    while Converged;
    /* M4                                  */
    do
        Learn $\mathbf{m}$ via Eq. (10);
    while Converged;
end
/* M6                                      */
Optimize $\Gamma^*(u, i | \Theta^*, \Phi)$ via Eq. (13);
```

2.3.5 Backbone. Recently, UltraGCN [27] achieves impressive performances. Though it is a GCN-base method, it serves as a MF-like linear model. Due to the effectiveness and simplicity, we select it as our backbone and make it adapt to multimedia recommendation. Formally, the predictive function of UltraGCN is defined as,

$$\Gamma(u, i, \mathbf{c}_i) = \Gamma(\mathbf{p}_u^{(t)}, \mathbf{p}_u^{(c)}, \mathbf{t}_i, \mathbf{c}_i) = \langle \mathbf{p}_u^{(t)}, \mathbf{t}_i \rangle + \langle \mathbf{p}_u^{(c)}, \mathbf{W} \cdot \mathbf{c}_i \rangle, \tag{14}$$

where $\mathbf{t}_i$ and $\mathbf{c}_i$ denote the collaborative and raw multimedia representations of item i, $\mathbf{p}_u^{(t)}$ and $\mathbf{p}_u^{(c)}$ denote the corresponding user representations, and $\mathbf{W}$ is a projection matrix to compress the dimension of the raw multimedia representation. UltraGCN pushes the representations to encode the user-item graph through the graph-based loss function,

$$\mathcal{L} = \mathcal{L}_O + \gamma_C \mathcal{L}_C + \gamma_I \mathcal{L}_I, \tag{15}$$

where γ_C and γ_I are hyper-parameters to balance the importance weights of these loss terms, and $\mathcal{L}_O$, $\mathcal{L}_C$ and $\mathcal{L}_I$ indicate the objective loss, the user-item constraint loss, and item-item constraint loss, respectively. The objective loss is,

$$\mathcal{L}_O = -\sum_{(u,i)\in\mathcal{R}} \log(\sigma(\Gamma(u,i))) - \sum_{(u,j)\in\mathcal{R}^-} \log(\sigma(-\Gamma(u,j))). \tag{16}$$

The constraint losses $\mathcal{L}_C$ and $\mathcal{L}_I$ are the keys to UltraGCN. They look toward the final stable status of graph-based approaches and try to append constraints to make the model learn the final representations directly. One is for the user-item interactive graph,

$$\mathcal{L}_C = -\sum_{(u,i)\in\mathcal{R}} \beta_{u,i} \log(\sigma(\Gamma(u,i))) - \sum_{(u,j)\in\mathcal{R}^-} \beta_{u,j} \log(\sigma(-\Gamma(u,j))), \tag{17}$$

where the fixed weight coefficients $\beta_{u,i}$ and $\beta_{u,j}$ are derived from the user-item interactive graph $\mathbf{R}$ by,

$$\beta_{u,i} = \frac{1}{d_u}\sqrt{\frac{d_u+1}{d_i+1}}, \tag{18}$$

where d_u and d_i denote the degrees of the corresponding nodes. Another constraint relies on an item-item correlation graph $\mathbf{G} = \mathbf{R}^T\mathbf{R}$, where $\mathbf{R}$ indicates the user-item interactive graph. Thus,

$$\mathcal{L}_I = -\sum_{(u,i)\in\mathcal{R}} \sum_{j\in S(i)} \omega_{i,j} \log(\sigma(\Gamma(u,j))), \tag{19}$$

where $S(i)$ indicates the adjacent item set of item i. The weight coefficient $\omega_{i,j}$ is computed by,

$$\omega_{i,j} = \frac{G_{i,j}}{g_i - G_{i,i}}\sqrt{\frac{g_i}{g_j}}, \quad g_i = \sum_k G_{i,k}, \tag{20}$$

where g_i and g_j denote the degrees of item i and item j in $\mathbf{G}$.

- **Improvement over UltraGCN.** UltraGCN is a powerful generic collaborative filtering method. Empirically, it gains competitive performance with the multimedia recommendations by simply concatenating the multimedia representations. Although it obtains strong collaborative embeddings, the drawback of the raw multimedia representations limits its upper bound. In contrast, InvRL maintains all the structures UltraGCN but uses **invariant item representations**. This change will significantly enhance the model for multimedia recommendation.

3 EXPERIMENTS

To evaluate the proposed InvRL, we conduct abundant experiments to answer the following research questions: **RQ1:** How does InvRL perform compared with the state-of-the-art multimedia recommendation methods? **RQ2:** How does the invariant mask $\mathbf{m}$ impact the results? **RQ3:** How does the environments $\mathcal{E}$ impact the results?

Table 1: The statistics on the datasets, Movielens, Tiktok, and Kwai. V, A, and T indicate the dimensions of visual, acoustic, and textual modalities, respectively.

Dataset	#Interactions	#Items	#Users	V	A	T
Movielens	1,239,508	5,986	55,485	2,048	128	100
Tiktok	726,065	76,085	36,656	128	128	128
Kwai	298,492	86,483	7,010	2,048	-	-

3.1 Experimental Settings

3.1.1 Datasets. Following the previous SOTA work on multimedia recommendation [38, 39], we conduct extensive experiments on three public datasets, Movielens, Tiktok, and Kwai. The statistics of the datasets are listed in Table 1.

Movielens[2] records the viewing history on movie information from the Movielens platform. In order to gain multimedia representations, the previous work [37] collected the corresponding trailers, titles, and descriptions and extracted the visual, acoustic, and textual semantic representations with pre-trained models (i.e. ResNet50 [9], VGGish [17], and Sentence2Vector [2]).

Tiktok[3] records the viewing history on micro-videos of the Tiktok platform. It provides visual, acoustic, and textual representations officially. As the original textual representations are given as sentences represented by one-hot words vectors, we take the sum of the word embeddings as the textual representations.

Kwai[4] records the viewing history on micro-videos of the Kwai platform. Each item corresponds to a 2048-d visual representation. Different from the above datasets, the items have no acoustic or textural representations. We follow the previous work [38, 39] and take the same interaction records for our experiments.

For fair comparisons, we split the dataset following previous work [38, 39] strictly. For each dataset, the interactions are put into the training/validation/testing sets with the ratio of 8:1:1. We tune the hyper-parameters according to the validation set and report the evaluation results on the testing set.

3.1.2 Evaluation Protocols. Following the previous work [38, 39], We score all the user-item interactions with trained models and rank them in descent order. For each user, we focus on the top-K items and compute the Precision@K (P@K), Recall@K (R@K), and Normalized Discounted Cumulative Gain (N@K) according to the observed interactions in the testing set. We take the average scores of all users to evaluate the trained model.

3.1.3 Baselines. To verify the effectiveness of InvRL, we compare it with the state-of-the-art multimedia recommendation methods. Generally, we adopt the baselines from three categories. **VBPR** [10], **DUIF** [8] and **CB2CF**[3] incorporate multimedia contents to original collaborative filtering method (CF), thus belonging to the multimedia CF (**M-CF**) category. **NGCF** [36], **DisenGCN** [25] and **MacridVAE** [26] belong to the generic neural CF (**G-NCF**) category. **MMGCN** [37], **HUIGN** [38], and **GRCN** [39] are multimedia-oriented NCF (**M-NCF**) Models. The performances of the above baselines are quoted from the previous work [38, 39].

[2] https://grouplens.org/datasets/movielens/.
[3] http://ai-lab-challenge.bytedance.com/tce/vc/.
[4] https://www.kuaishou.com/activity/uimc/.

UltraGCN [27] analyzes the message passing through the graph collaborative filtering and simplifies the message passing procedure as a regularization term. Due to its strong performances on collaborative filtering, we select it as our backbone. To adapt to the multimedia tasks, we concatenate the collaborative embeddings and content representations as the item representations.

We adopt InvRL to indicate our framework instantiated over UltraGCN. We implement UltraGCN and InvRL with Pytorch[5].

3.1.4 Parameter Settings.

More specifically, we empirically took the Adam [19] as the optimizer. We set the batch size as 512 and fix the embedding dimension to 64. We individually tune the learning rates and regularization factors for id-specific embeddings and other parameters. In many cases, we adopt regularization factors with 10^{-4} weight for id-specific parameters and tune them in $\{1, 0.1, 0.01, 0.001, 0\}$ for other parameters, and we set the learning rate to 10^{-3} for all parameters. The number of environments $|\mathcal{E}|$ is tuned in $\{1, 5, 10, 20, 30\}$, α and λ in Equation 10 are respectively selected in $\{1, 0.5, 0.1\}$, and the learning rate for $\mathbf{m}$ in mask generation module is searched in $\{0.01, 0.001, 0.0001\}$. γ_C and γ_I are tuned in $\{2, 1, 0.1, 0.01, 0\}$. The iteration parameter T is initially set as 5. Environment partition model and mask generation model are trained for 20 and 40 epochs respectively, and the final predictive model is trained for 500 epochs. We choose the model according to the validation scores and report the corresponding testing scores.

3.2 Performance Comparison (RQ1)

We list the overall performance comparison among the methods in Table 2. We have the following observations,

• The neural collaborative filtering (NCF) approaches mostly outperform the collaborative filtering (CF) approaches. This illustrates the effectiveness of modeling high-order correlations among users and items. Also, the worse performances of DUIF reveal the impact of collaborative support. In addition, the M-NCF approaches mostly outperform the G-NCF approaches, which verifies the necessity of multimedia-specific adaptation for the multimedia recommendation. GRCN yields the best performances in the NCF-based category since it incorporates both user intents and item contents significantly. Accordingly, an effective multimedia recommendation model should incorporate the power from both the user behaviors and item contents.

• Our backbone UltraGCN is also a generic graph-based CF method. Although UltraGCN seems an MF-based model and just simply incorporates the multimedia contents, it outperforms other multimedia recommendation baselines by a large margin. The impressive performance reveals that UltraGCN can model the GCF embeddings through the constraint losses, thus exploiting the inherent collaborative information. However, there is no special design for the multimedia content, we believe that there is more space to enhance its performance.

• InvRL achieves the best performances on all three datasets. It outperforms UltraGCN by 3.78%, 8.71% and 7.07% in Movielens, Tiktok, and Kwai, respectively. Compared with UltraGCN, the final model of InvRL adopts the same predictive function and the same training target, except for the use of the content representation (as shown in Section 2.3.4). To alleviate the impact of spurious correlations, the InvRL takes in the invariant item representations which are generated via a learned invariant mask. Therefore, we believe that the significant improvements are caused by the constraint of the mask. This verifies the effectiveness of our framework.

• The improvement ratios rely on the properties of the datasets. The improvements of InvRL on Tiktok are greater than 7%, which is much larger than the other two datasets. Compared with Kwai, Tiktok consists of three modalities, while Kwai has only one. More modalities provide more perspectives, thus making it easier to engineer invariant representations. Compared with Movielens, Tiktok has much fewer interactions for each item. Due to the deficiency of collaborative information, they gain more improvements from the explorations of multimedia representations. This further reveals the significance of our work to the multimedia recommendation domain since the items usually have a very small number of interactions.

To emphasize the improvements of InvRL, we make another comparison between InvRL and the backbone UltraGCN, by listing their top-K scores. Figure 3 exhibits the curves of NDCG scores *w.r.t.* the three datasets. A salient observation is that InvRL consistently outperforms UltraGCN for different K. The scores of P@K and R@K perform similarly (unshown here due to the page limitation). This reflects the improvements of InvRL enhance the overall recommending results.

3.3 Study of the Mask (RQ2)

InvRL utilizes an invariant representation indicator which is a mask vector to separate the raw multimedia representations. Different from the previous work HRM [23], we adopt a float vector as the substitute for the binary vector. Correspondingly, we revise the L_0 norm to the L_2 norm in Equation 10. Here, we conduct two experiments to verify the effectiveness of our settings and demonstrate the power of the mask for multimedia recommendation.

Table 3 demonstrates the variants of the mask. B and F indicate the binary mask and float mask, respectively. L_0 and L_2 indicate two regularization terms for the masks, respectively. From the performances, we have two observations,

(1) The results of float vectors always outperform those of binary vectors. This verifies that each dimension in the continuous representations is significant for the recommendation. The binary setting may remove some useful dimensions even though they are less important.

(2) The L_0 norm always impacts the performances negatively compared with L_2. The major reason is that the L_0 norm is a sparse constraint. It may also suppress some valuable dimensions, thus reducing the testing scores.

Accordingly, we suggest to use the float mask constrained by the L_2 norm in Equation 10.

To further comprehend how $\mathbf{m}$ works, we visualize the dimensional values of $\mathbf{m}$ in Figure 4. According to the value distribution, we observe that the mask differentiates representations automatically. The three modalities in Movielens and Tiktok perform different patterns. This verifies the drawback of the native use of the raw representations. Moreover, the weight distribution in a single modality also reveals the different contributions of different dimensions. This illustrates the necessity and the feasibility of capturing the invariant part with InvRL.

[5] https://pytorch.org/

Table 2: Performances. The bold scores indicate the best performance. The underlined scores indicate the second-best performance. M-CF, G-NCF and M-NCF indicate the multimedia CF, generic NCF and multimedia NCF, respectively.

Category	Methods	Movielens			Tiktok			Kwai		
		P@10	R@10	N@10	P@10	R@10	N@10	P@10	R@10	N@10
M-CF	VBPR	0.0512	0.1990	0.2261	0.0118	0.0628	0.0574	0.0132	0.0410	0.0702
	DUIF	0.0538	0.2167	0.2341	0.0087	0.0483	0.0434	0.0095	0.0258	0.0521
	CB2CF	0.0548	0.2265	0.2505	0.0109	0.0642	0.0613	0.0132	0.0407	0.0700
G-NCF	NGCF	0.0547	0.2196	0.2342	0.0135	0.0780	0.0661	0.0118	0.0402	0.0699
	DisenGCN	0.0555	0.2222	0.2401	0.0145	0.0760	0.0639	0.0127	0.0403	0.0683
	MacridVAE	0.0576	0.2286	0.2437	0.0152	0.0813	0.0686	0.0130	0.0405	0.0685
M-NCF	MMGCN	0.0581	0.2345	0.2517	0.0144	0.0808	0.0674	0.0120	0.0398	0.0681
	HUIGN [38]	0.0619	0.2522	0.2677	0.0164	0.0884	0.0769	0.0140	0.0434	0.0778
	GRCN [39]	0.0639	0.2569	0.2754	0.0195	0.1048	0.0938	0.0168	0.0492	0.0864
Backbone	UltraGCN	0.0630	0.2569	0.2709	0.0182	0.0982	0.0876	0.0174	0.0497	0.0922
Ours	InvRL	**0.0652**	**0.2672**	**0.2813**	**0.0196**	**0.1079**	**0.0951**	**0.0187**	**0.0532**	**0.0984**
%Impv. over Backbone		3.49%	4.01%	3.84%	7.69%	9.87%	8.56%	7.47%	7.04%	6.72%

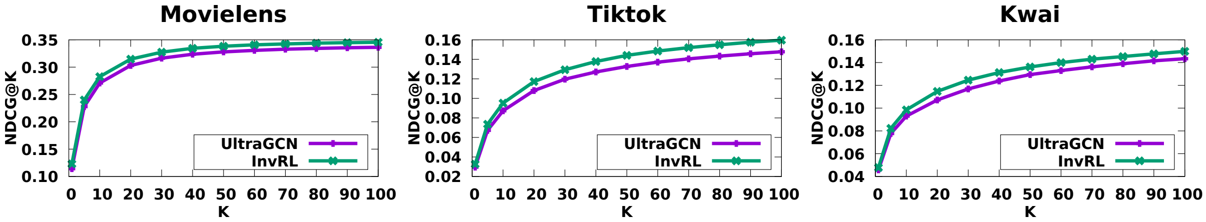

Figure 3: The Comparison between UltraGCN and InvRL *w.r.t.* NDCG@K. The situation that InvRL outperforms UltraGCN consistently reveals the overall improvements over the recommending results.

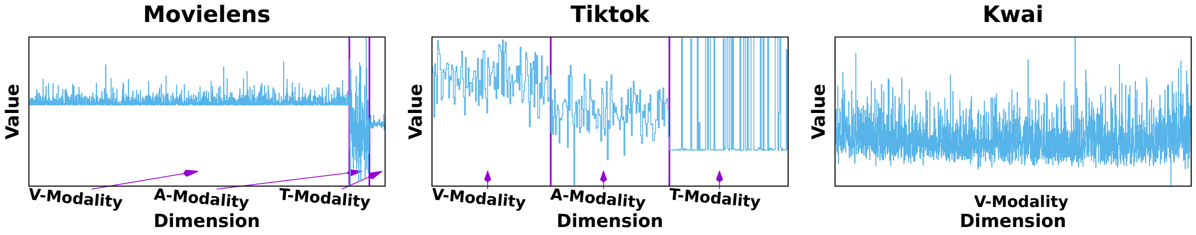

Figure 4: Visualization of the masks. The weight distributions from different modalities perform different patterns.

Table 3: The impact of m-related settings. B and F indicate the binary mask and float mask, respectively. L_0 and L_2 indicate two regularization terms for the masks, respectively

Dataset	m-Type	P@10	R@10	N@10
Movielens	B+L_0	0.0640	0.2618	0.2762
	B+L_2	0.0646	0.2639	0.2776
	F+L_0	0.0649	0.2658	0.2798
	F+L_2	**0.0652**	**0.2672**	**0.2813**
Tiktok	B+L_0	0.0191	0.1070	0.0940
	B+L_2	0.0194	0.1068	0.0944
	F+L_0	0.0194	0.1068	0.0933
	F+L_2	**0.0196**	**0.1079**	**0.0951**
Kwai	B+L_0	0.0180	0.0530	0.0964
	B+L_2	0.0184	0.0526	0.0972
	F+L_0	0.0185	0.0523	0.0968
	F+L_2	**0.0187**	**0.0532**	**0.0984**

3.4 Study of the Environments (RQ3)

The environments correspond to the spurious correlations in user-item interactions. We conduct two experiments to verify the existence and separability of the environments and explore the empirical experiences for further applications.

In the first experiment, we record the ratio of moved interactions during the process of module M3. Specifically, after the environments are updated, the interactions are partitioned again. According to Equation 7, if an interaction still fits the current environment, it would not be moved, otherwise, we move it to the new environment and count it as moved interaction. By repeating the environment partition several times, we gain the curves shown in Figure 5. All the three curves converge after being repeated 20 times. Only a small number of interactions would be moved then. Thus the environments come to stable. This reveals the separability of the environments thus InvRL can model the spurious correlations correctly. In addition, the curve implies the appropriate loop count in module M3 of Algorithm 1.

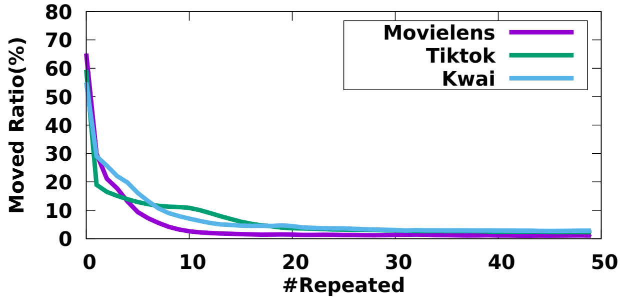

Figure 5: The ratio of moved interactions during the progress of environment partitions.

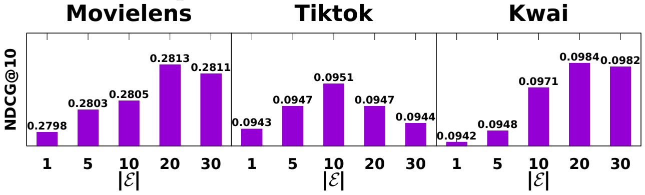

Figure 6: The impact of $|\mathcal{E}|$, *i.e.,* the number of environments. Sufficient environments mostly lead to better performances.

To explore the partition of the environments, we conduct multiple experiments with different $|\mathcal{E}|$, which indicates the number of environments. As shown in Figure 6, with the increment of $|\mathcal{E}|$, the performances have an ascending trend. More environments support more detailed interaction partitions, thus leading to better performances. This also implies the existence of the spurious correlations from another perspective. However, considering the expend of computational resources, we just test $|\mathcal{E}|$ with no larger than 30.

4 RELATED WORK

Collaborative Filtering (CF) is a crucial method for recommendation. It analyzes the observed interactions between users and items and capture the collaborative embeddings. The fundamental method is matrix factorization [15] that assumes users and items have much similar properties thus representing them with low-rank embeddings. FISM [18] assumes that users would like the things similar to the interacted items, thus represents users with interacted items. SVD++ [20] incorporates the two angles and achieves impressive performance. Subsequently, many works explore the improvements on the fundamental CF methods. BPR [29] proposes a pair-wise loss to describe recommendation tasks in a ranking view. ALS [14] assigns weights to each interaction. APR [12] utilizes adversarial samples to train robust models. For better recommendation, many novel techniques are introduced, such as causality [16, 34, 43], knowledge graphs [35, 41], neural networks [7, 13], etc.

Neural structures are widely used to exploit the inherent collaborative properties. NCF [13] feeds the user and item embeddings into a two-branch multi-layer perception network to predict the user preference. ConvNCF [7] models the cross-dimension information and predicts with a convolutional network. Liang *et al.* [21] proposed a VAE-based CF method to reconstruct the ideal interaction matrix. It is notable that graph neural networks perform very well for the recommending tasks since the user-item interaction matrices naturally describe a bipartite graph. Therefore, NGCF [36] introduces the graph convolutional network (GCN) to model the high-hop neighbor information. Similarly, PinSage [41] adopts random walk and GCN simultaneously to traverse the graph. KGAT [35] focuses on extra knowledge graph and also uses the random walk manner to seek the relations. Lightgcn [11] expands GCN in recommending perspective and proposes a fast and powerful model. The most impressive method is UltraGCN [27], which learns the graph-enhanced representation directly. Due to the simplicity and effectiveness, we select it as the backbone.

Multimedia recommendation methods mostly correspond to some CF methods. VBPR [10], known as the benchmark of multimedia recommendation, follows the MF structure but represents items with collaborative embedding and content representations simultaneously. DUIF [8] is another MF-based method that represents the items with their content representations only. As GCN-based CF performs well in many cases, many works explore multimedia-oriented GCN. MMGCN [37] proposes a multi-graph neural network to process and integrate the multi-modal representations. MGAT [32] adopts an attentive mechanism to enhance the multi-modal representation integration. MKGAT [31] utilizes the attentive mechanism to generate representations across modals. DisenGCN [25] disentangles the latent factors of the interactions. MacridVAE [26] forces each dimension of the representations to independently reflect an isolated low-level factor. Compared with the ordinary CF models, multimedia recommendation explores the use of multimedia representations, *a.k.a.* , the generation of invariant item representations.

There are two directions to capture invariant representations, 1) to engineer preference-aware representations, and 2) to block the spurious correlations. For the first direction, Yu *et al.* [42] adopted an aesthetic model to provide aesthetic evaluation as they aimed to address the fashion recommendation task. Deepstyle [24] separates the clothes representations into category and style parts, since they argue that the category information may cause spurious correlation. AMAE [30] proposes an attention-based model to extract profitable representations from music-related information. For the second direction, Wang *et al.* [33] differentiated the real 'like' behavior from 'click' via causal reference, but more labeled data are required. Qiu*et al.* [28] debiases the visual representations from a causal perspective. However, These approaches rely on domain specific analysis and design, lack of a generic mechanism to eliminate the influence of the spurious correlation. Therefore, we propose InvRL to exploit an effective way.

5 CONCLUSION

In this paper, we proposed an invariant representation learning framework (InvRL) for multimedia recommendation. We adopted heterogeneous environments to indicate the spurious correlations and learn invariant item representations across the environments. According to the invariant representations, we finally trained a recommender model that achieves the best performances in the experiments. This work is a bold attempt on alleviating the spurious correlations from the multimedia contents. In future, we will make more efforts in exploring the user behaviors and the item multimedia contents, including but not limited to a new representation generative mechanism, a more practical and efficient representation separator, *etc.*

REFERENCES

[1] Kartik Ahuja, Karthikeyan Shanmugam, Kush R. Varshney, and Amit Dhurandhar. 2020. Invariant Risk Minimization Games. *arXiv preprint arXiv:2002.04692* (2020). arXiv:2002.04692 http://arxiv.org/abs/2002.04692

[2] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *5th International Conference on Learning Representations*.

[3] Oren Barkan, Noam Koenigstein, Eylon Yogev, and Ori Katz. 2019. CB2CF: A neural multiview content-to-collaborative filtering model for completely cold item recommendations. In *13th ACM Conference on Recommender Systems*. 228–236. https://doi.org/10.1145/3298689.3347038

[4] Zhiyong Cheng, Xiaojun Chang, Lei Zhu, Rose C. Kanjirathinkal, and Mohan Kankanhalli. 2019. MMalfM: Explainable recommendation by leveraging reviews and images. *ACM Transactions on Information Systems* 37, 2 (2019), 1–28. https://doi.org/10.1145/3291060 arXiv:1811.05318

[5] Jacob Devlin, Ming Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. *2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* 1 (2019), 4171–4186. arXiv:1810.04805

[6] Yujuan Ding, Yunshan Ma, Wai Keung Wong, and Tat Seng Chua. 2022. Modeling Instant User Intent and Content-Level Transition for Sequential Fashion Recommendation. *IEEE Transactions on Multimedia* 24 (2022), 2687–2700. https://doi.org/10.1109/TMM.2021.3088281

[7] Xiaoyu Du, Xiangnan He, F. Yuan, Jinhui Tang, Zhiguang Qin, and Tat Seng Chua. 2019. Modeling embedding dimension correlations via convolutional neural collaborative filtering. *ACM Transactions on Information Systems* 37, 4 (2019), 1–22. https://doi.org/10.1145/3357154 arXiv:1906.11171

[8] Xue Geng, Hanwang Zhang, Jingwen Bian, and Tat Seng Chua. 2015. Learning image and user features for recommendation in social networks. In *Proceedings of the IEEE International Conference on Computer Vision*, Vol. 2015 Inter. 4274–4282. https://doi.org/10.1109/ICCV.2015.486

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 2016-December. 770–778. https://doi.org/10.1109/CVPR.2016.90 arXiv:1512.03385

[10] Ruining He and Julian McAuley. 2016. VBPR: Visual Bayesian personalized ranking from implicit feedback. In *30th AAAI Conference on Artificial Intelligence*, Vol. 30. 144–150. https://doi.org/10.1609/aaai.v30i1.9973 arXiv:1510.01784

[11] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yong Dong Zhang, and Meng Wang. 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 639–648. https://doi.org/10.1145/3397271.3401063 arXiv:2002.02126

[12] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat Seng Chua. 2018. Adversarial personalized ranking for recommendation. In *41st International ACM SIGIR Conference on Research and Development in Information Retrieval*. 355–364. https://doi.org/10.1145/3209978.3209981 arXiv:1808.03908

[13] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat Seng Chua. 2017. Neural collaborative filtering. In *26th International World Wide Web Conference*. 173–182. https://doi.org/10.1145/3038912.3052569 arXiv:1708.05031

[14] Xiangnan He, Jinhui Tang, Xiaoyu Du, Richang Hong, Tongwei Ren, and Tat Seng Chua. 2020. Fast matrix factorization with nonuniform weights on missing data. *IEEE Transactions on Neural Networks and Learning Systems* 31, 8 (2020), 2791–2804. https://doi.org/10.1109/TNNLS.2018.2890117 arXiv:1811.04411

[15] Xiangnan He, Hanwang Zhang, Min Yen Kan, and Tat Seng Chua. 2016. Fast matrix factorization for online recommendation with implicit feedback. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 549–558. https://doi.org/10.1145/2911451.2911489 arXiv:1708.05024

[16] Xiangnan He, Yang Zhang, Fuli Feng, Chonggang Song, Lingling Yi, Guohui Ling, and Yongdong Zhang. 2022. Addressing Confounding Feature Issue for Causal Recommendation. *arXiv preprint arXiv:2205.06532* (2022). arXiv:2205.06532 http://arxiv.org/abs/2205.06532

[17] Shawn Hershey, Sourish Chaudhuri, Daniel P.W. Ellis, Jort F. Gemmeke, Aren Jansen, R. Channing Moore, Manoj Plakal, Devin Platt, Rif A. Saurous, Bryan Seybold, Malcolm Slaney, Ron J. Weiss, and Kevin Wilson. 2017. CNN architectures for large-scale audio classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 131–135. https://doi.org/10.1109/ICASSP.2017.7952132 arXiv:1609.09430

[18] Santosh Kabbur, Xia Ning, and George Karypis. 2013. FISM: Factored item similarity models for Top-N recommender systems. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Vol. Part F128815. 659–667. https://doi.org/10.1145/2487575.2487589

[19] Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations* (2015). arXiv:1412.6980

[20] Yehuda Koren. 2008. Factorization meets the neighborhood: A multifaceted collaborative filtering model. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 426–434. https://doi.org/10.1145/1401890.1401944

[21] Dawen Liang, Rahul G. Krishnan, Matthew D. Hoffman, and Tony Jebara. 2018. Variational autoencoders for collaborative filtering. In *Proceedings of the World Wide Web Conference*. 689–698. https://doi.org/10.1145/3178876.3186150 arXiv:1802.05814

[22] Fan Liu, Zhiyong Cheng, Changchang Sun, Yinglong Wang, Liqiang Nie, and Mohan Kankanhalli. 2019. User diverse preference modeling by multimodal attentive metric learning. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1526–1534. https://doi.org/10.1145/3343031.3350953 arXiv:1908.07738

[23] Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. 2021. Kernelized Heterogeneous Risk Minimization. In *Advances in Neural Information Processing Systems*, Vol. 26. PMLR, 21720–21731. arXiv:2110.12425

[24] Qiang Liu, Shu Wu, and Liang Wang. 2017. Deepstyle: Learning user preferences for visual recommendation. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '17)*. 841–844. https://doi.org/10.1145/3077136.3080658

[25] Jianxin Ma, Peng Cui, Kun Kuang, Xin Wang, and Wenwu Zhu. 2019. Disentangled graph convolutional networks. In *36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 2019-June)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 7454–7463.

[26] Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. 2019. Learning disentangled representations for recommendation. *Advances in Neural Information Processing Systems* 32 (2019). arXiv:1910.14238

[27] Kelong Mao, Jieming Zhu, Xi Xiao, Biao Lu, Zhaowei Wang, and Xiuqiang He. 2021. UltraGCN: Ultra Simplification of Graph Convolutional Networks for Recommendation. In *International Conference on Information and Knowledge Management, Proceedings*. 1253–1262. https://doi.org/10.1145/3459637.3482291 arXiv:2110.15114

[28] Ruihong Qiu, Sen Wang, Zhi Chen, Hongzhi Yin, and Zi Huang. 2021. CausalRec: Causal Inference for Visual Debiasing in Visually-Aware Recommendation. In *Proceedings of the 29th ACM International Conference on Multimedia*. 3844–3852. https://doi.org/10.1145/3474085.3475266 arXiv:2107.02390

[29] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence* (2009), 452–461. arXiv:1205.2618

[30] Tiancheng Shen, Jia Jia, Yan Li, Hanjie Wang, and Bo Chen. 2020. Enhancing Music Recommendation with Social Media Content: An Attentive Multimodal Autoencoder Approach. In *Proceedings of the International Joint Conference on Neural Networks*. IEEE, 1–8. https://doi.org/10.1109/IJCNN48605.2020.9206894

[31] Rui Sun, Xuezhi Cao, Yan Zhao, Junchen Wan, Kun Zhou, Fuzheng Zhang, Zhongyuan Wang, and Kai Zheng. 2020. Multi-modal Knowledge Graphs for Recommender Systems. *International Conference on Information and Knowledge Management, Proceedings* (2020), 1405–1414. https://doi.org/10.1145/3340531.3411947

[32] Zhulin Tao, Yinwei Wei, Xiang Wang, Xiangnan He, Xianglin Huang, and Tat Seng Chua. 2020. MGAT: Multimodal Graph Attention Network for Recommendation. *Information Processing and Management* 57, 5 (2020), 102277. https://doi.org/10.1016/j.ipm.2020.102277

[33] Wenjie Wang, Fuli Feng, Xiangnan He, Hanwang Zhang, and Tat Seng Chua. 2021. Clicks can be Cheating: Counterfactual Recommendation for Mitigating Clickbait Issue. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1288–1297. https://doi.org/10.1145/3404835.3462962 arXiv:2009.09945

[34] Wenjie Wang, Xinyu Lin, Fuli Feng, Xiangnan He, Min Lin, and Tat Seng Chua. 2022. Causal Representation Learning for Out-of-Distribution Recommendation. In *Proceedings of the ACM Web Conference 2022*. 3562–3571. https://doi.org/10.1145/3485447.3512251

[35] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat Seng Chua. 2019. KGAT: Knowledge graph attention network for recommendation. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 950–958. https://doi.org/10.1145/3292500.3330989 arXiv:1905.07854

[36] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 165–174. https://doi.org/10.1145/3331184.3331267 arXiv:1905.08108

[37] Yinwei Wei, Xiangnan He, Xiang Wang, Richang Hong, Liqiang Nie, and Tat Seng Chua. 2019. MMGCN: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the 27th ACM International Conference on Multimedia*. 1437–1445. https://doi.org/10.1145/3343031.3351034

[38] Yinwei Wei, Xiang Wang, Xiangnan He, Liqiang Nie, Yong Rui, and Tat Seng Chua. 2022. Hierarchical User Intent Graph Network for Multimedia Recommendation. *IEEE Transactions on Multimedia* 24 (2022), 2701–2712. https://doi.org/10.1109/TMM.2021.3088307 arXiv:2110.14925

[39] Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, and Tat Seng Chua. 2020. Graph-Refined Convolutional Network for Multimedia Recommendation with Implicit Feedback. *Proceedings of the 28th ACM International Conference on Multimedia* (2020), 3541–3549. https://doi.org/10.1145/3394171.3413556 arXiv:2111.02036

[40] Le Wu, Lei Chen, Richang Hong, Yanjie Fu, Xing Xie, and Meng Wang. 2020. A Hierarchical Attention Model for Social Contextual Image Recommendation. *IEEE Transactions on Knowledge and Data Engineering* 32, 10 (2020), 1854–1867. https://doi.org/10.1109/TKDE.2019.2913394 arXiv:1806.00723

[41] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 974–983. https://doi.org/10.1145/3219819.3219890 arXiv:1806.01973

[42] Wenhui Yu, Huidi Zhang, Xiangnan He, Xu Chen, Li Xiong, and Zheng Qin. 2018. Aesthetic-based clothing recommendation. In *Proceedings of the World Wide Web Conference*. 649–658. https://doi.org/10.1145/3178876.3186146 arXiv:1809.05822

[43] Xinyuan Zhu, Yang Zhang, Fuli Feng, Xun Yang, Dingxian Wang, and Xiangnan He. 2022. Mitigating Hidden Confounding Effects for Causal Recommendation. *arXiv preprint arXiv:2205.07499* (2022). arXiv:2205.07499 http://arxiv.org/abs/2205.07499