Semantics-Guided Disentangled Learning for Recommendation

Dianer Yu[1], Qian Li[2✉], Xiangmeng Wang[1], Zhichao Wang[3], Yanan Cao[4], and
Guandong Xu[1✉]

[1] Data Science and Machine Intelligence Lab, School of Computer Science,
University of Technology Sydney, Sydney, Australia
`{Dianer.Yu-1,Xiangmeng.Wang}@student.uts.edu.au, Guandong.Xu@uts.edu.au`
[2] School of Electrical Engineering, Computing and Mathematical Sciences
Curtin University, Perth, Australia, `qli@curtin.edu.au`
[3] University of New South Wales, Australia, `zchaoking@gmail.com`
[4] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
`caoyanan@iie.ac.cn`

Abstract. Although traditional recommendation methods trained on observational interaction information have engendered a significant impact in real-world applications, it is challenging to disentangle users' true interests from interaction data. Recent disentangled learning methods emphasize on untangling users' true interests from historical interaction records, which however overlook auxiliary information to correct bias. In this paper, we design a novel method called **SeDLR** (**Se**mantics **D**isentangled **L**earning **R**ecommendation) to bridge this gap. Particularly, by leveraging rich heterogeneous information networks (HIN), SeDLR is capable of untangling high-order user-item relationships into multiple independent components according to their semantic user intents. In addition, SeDLR offers reliable explanations for the disentangled graph embeddings by the designed Monte Carlo edge-drop component. Finally, we conduct extensive experiments on two benchmark datasets and achieve state-of-the-art performance compared against recent strong baselines.

Keywords: Semantic-aware Representation · Disentangled Learning · Monte Carlo Edge-drop · Explainable Recommendation.

1 Introduction

Recommendation systems (RS) have become popular personalization tools to assist users in sorting through the ever-growing corpus of content and discovering contents in which they would be interested [11, 3, 22, 13]. Early work mainly used collaborative filtering methods to simply learn user/item ID representation based on historical interactions [23, 20, 1]. More effective methods exploit interaction as graph-structured data and aggregate feature information from high-order neighborhoods using neural networks [7, 20, 14]. Despite effectiveness, modeling

user-item relationships via embedding functions fails to differentiate user intents on different items, which could easily lead to suboptimal representations [21, 9, 5]. The disentangled learning emerges as the state-of-the-art and aims to explore the diverse user-item relationships and learn disentangled representation for users' true interests[12, 21, 11, 4, 25].

The principal motivation of disentangled learning is to separate users' intents behind each interaction in order to achieve a robust recommendation. Although disentangled learning has made promising improvements for distilling users' intents, a deficiency is that they emphasize historical interaction records and overlook auxiliary information to correct bias in the recommendation. As shown in Figure 1, there are four interactions between u_1 and movies (i.e., i_1, i_2, i_3 and i_4). With context information of user and item, we may infer that u_1 prefers to watch a movie with type and director. More importantly, the interaction between u_1 and i_4 might be due to the conformity bias that u_1 tends to watch a movie i_4 that is strongly recommended by a friend u_2, even if this goes against u_1's own preference. Merely using interactions without contextual information fails to capture users' pure interests that are independent of conformity. Therefore, exploiting the contextual information of users (e.g., social relationship) and movie (i.e., director and type) is crucial for distinguishing the conformity bias from users' true interests.

Towards this end, we empower disentangled learning with contextual information, with the aim of discovering users' true interests from the biased interactions and offering explainable recommendations. Overall, the three main contributions of this work are summarized as followings:
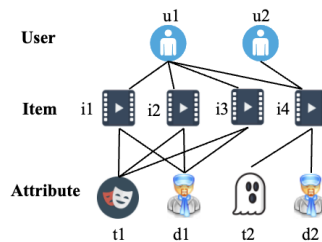

Fig. 1. An illustration of interactions between users and items with contexts.

- To the best of our knowledge, we are the first to incorporate heterogeneous information networks (HIN) into disentangled learning. Our SeDLR model can exploit high-order user-item relationships at the finer granularity and learn disentangled representations towards different semantic-aware aspects.
- We design a Monte Carlo edge-drop strategy, which modifies the HIN structure and drops users' intents-irrelevant semantic information, with the aim of facilitating the explainability of our SeDLR model.
- We conduct extensive experiments on two benchmark datasets to show the superiority and explainability of our SeDLR model.

2   Preliminary and Related Work

In this section, we will introduce recent works that are highly related to ours includes HIN-based learning, graph-based entangled learning and disentangled learning for the recommendation.

Heterogeneous information networks (HIN) include multiple node types and connection relationships, which can flexibly use rich objects and information to model heterogeneous data effectively [18]. HIN enhanced methods leverage meta-path based social relationships derived from rich HIN information, which greatly improve the Top-K recommendation performance. Many HIN-based recommendations have proven the effectiveness of using HIN. For example, IF-BPR [24] propose meta-path based social relations derived from a HIN, then capture the similarity between users for the recommendation. While MCRec [8] uses rich meta-path context representation and attention mechanism.

Graph-based entangled methods learn user/item embeddings by linearly propagating with neighborhood aggregation in the Graph Convolution Network (GCN) componentsuch as NeuMF [7] and NGCF [20]. NeuMF [7] combines traditional matrix factorization and neural network, which can extract low and high dimensional features at the same time, then concatenate multiple neural network layers with matrix factorization layer to gain the final likelihood score [10]. While NGCF [20] refine the embedding vector from high-order connection information, and integrates by three Graph Neural Network (GNN) layers, then trains by optimizing losses to gain the affinity score of the pair of user and item.

Disentanglement recommendation methods learn users' intents by disentangling users' latent factors, which is more effective to recommend items by knowing the intent rather than the historical records [21]. For instance, DGCF [21] is a state-of-the-art disentanglement recommendation method, which disentangles latent factors of user intents by the neighbor routing and embedding propagation, then applies an independent module to separate intents. M-VAE [16] achieves the macroscopic entanglement by inferring the high-level concepts associated with user intentions, and simultaneously captures user preferences for different items. However, neither M-VAE nor DGCF is able to associate learned intent with real-world users' aspects which can be seen as pre-defined intent.

3   Methodology

The architecture of the proposed SeDLR's framework is shown in Figure 2. Our method takes the holistic user-item interaction graph with a HIN as the input, and passes through a graph disentangling network (top-left) to divide the holistic interaction graph into q intent-aware sub-graphs for learning the separated user intent representations, while the HIN embedding network (bottom-left) leverages meta-path schemes retained in the HIN to construct expressive representations of context (i.e., aspects). The learned context representations are then incorporated into user intent representations to derive the semantic-aware intent representation for the later recommendation task. Finally, to better explain

the disentangled learning based recommendation, we use Monte Carlo edge-drop strategy to select the important aspects as explanations (right).
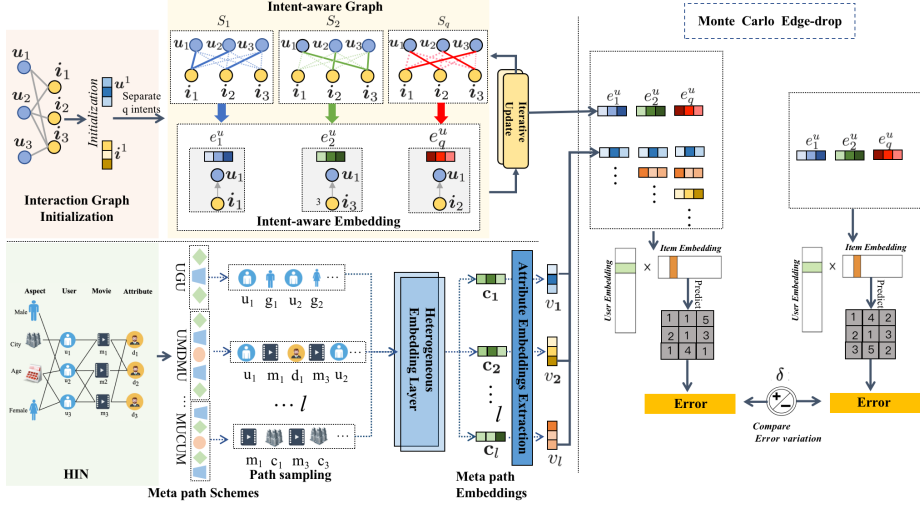

Fig. 2. The overview of the proposed SeDLR framework.

3.1   Graph Disentangling for Users Intents

Our first target is to disentangle q intents of the user as initialization, hence we divide user/item embedding into q chunks and associate each with a potential intent as follows:

$$\boldsymbol{u} = (\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_q), \quad \boldsymbol{i} = (\boldsymbol{i}_1, \boldsymbol{i}_2, \ldots, \boldsymbol{i}_q) \tag{1}$$

where $\boldsymbol{u}_q$ and $\boldsymbol{i}_q$ illustrate chunked representation for q-th intent on interaction of user/item. Additionally, we employ random initialization for each chunk representation to ensure the difference before the training stage. We then adopt a score vector to explore the relationships between intent and interaction as:

$$S(u, i) = (S_1(u, i), S_2(u, i), \cdots, S_q(u, i)) \tag{2}$$

where $S_q(u, i)$ represents the score vector over q-th intent on interaction, which is the possibility of adopting interaction is due to q-th intent. Accordingly, a set of score vectors can be initialized as the same values to indicate the same contribution on interaction before training. Hence, this score vector can be seen as an adjacency matrix for an intent-aware graph.

Next, we design a graph disentangling layer to explore valuable information via the high-order connectivity, we employ a graph disentangling layer that consists of embedding propagation mechanisms and neighbor routing as follows:

$$e_q^{u(1)} = g\left(\boldsymbol{u}_q, \{\boldsymbol{i}_q \mid i \in \mathcal{N}_u\}\right) \tag{3}$$

where $e_q^{u(1)}$ collects information from neighbors of u, and index 1 represents the first-order neighbor. $\mathcal{N}_u$ is the historical interacted items, more formally, it is the first-hop neighbor. We then perform an iterative update rule which is to update the intent-aware embedding by embedding propagation in the intent-aware graph, then use it to refine the graph. Consequently, score vector S_q for each interaction after n iteration can be generated. To get its distribution across all intents, we normalized by softmax as:

$$\tilde{S}_q^n(u, i) = \frac{\exp S_q^n(u, i)}{\sum_{q'=1}^{q} \exp S_{q'}^n(u, i)} \tag{4}$$

to illustrate the importance of each intent. Accordingly, we can obtain normalized adjacency matrix $\tilde{S}_q^n$ for each intent. The Laplacian matrix of $\tilde{S}_q^n$ is adopted as:

$$\mathcal{M}_q^n(u, i) = \frac{\tilde{S}_q^n(u, i)}{\sqrt{D_q^n(u) \cdot D_q^n(i)}} \tag{5}$$

where $D_q^n(u) = \sum_{i' \in \mathcal{N}_u} \tilde{S}_q^n(u, i')$ and $D_q^n(i) = \sum_{u' \in \mathcal{N}_i} \tilde{S}_q^n(u', i)$ are the degrees of u and i, respectively. Besides, the embedding propagation for each graph can encode the information influenced to the interaction and the sum aggregator is defined as:

$$\boldsymbol{u}_q^n = \sum_{i \in N_u} \mathcal{M}_q^n(u, i) \cdot \boldsymbol{i}_q^0 \tag{6}$$

where $\boldsymbol{u}_q^n$ illustrates the sum of historical items and importance weighting in q-th aspect at n-th iteration, and $\boldsymbol{i}_q^0$ is the input representation for the historical item. It can temporarily memorize the information collected from neighbors $\mathcal{N}_u$.

Thereafter, we iteratively the update intent-aware graph. Intuitively, interacted items driven by the same intent tend to have similar chunked representations, encouraging stronger relationships between them can achieve this purpose. Thus, we iteratively update the interaction score vector $S_q^n(u, i)$ to adjust the degree of u and neighbor i as follows:

$$S_q^{n+1}(u, i) = S_q^n(u, i) + \boldsymbol{u}_q^{n\top} \tanh\left(\boldsymbol{i}_q^0\right) \tag{7}$$

where $\boldsymbol{u}_q^{n\top} \tanh\left(\boldsymbol{i}_q^0\right)$ represents the affinity between $\boldsymbol{u}_q^n$ and $\boldsymbol{i}_q^0$ in Eq.(6), while $tanh$ is a nonlinear activation function can improve the representation ability. Finally, output a graph disentangling layer after n iterations, that contains disentangled representation $e_q^{u(1)} = \boldsymbol{u}_q^n$ and intent-aware graph $\tilde{S}_q^n$.

We then combine multiple layers to gather rich semantics from high-order connectivity. While the first-order neighbors have been used above, hence we can stack r layers to obtain influence signals from r-th high-order connectivity as:

$$e_q^{u(r)} = g\left(e_q^{u(r-1)}, \left\{e_q^{i(r-1)} \mid i \in \mathcal{N}_u\right\}\right) \tag{8}$$

where $e_q^{u\,(r-1)}$ and $e_q^{i\,(r-1)}$ serve as the representations of u and i on q-th intent which save the propagated information from $(r\text{-}1)$-hop neighbors. Every disentangled representation is associated with explanatory graph serve as weighted adjacency matrix $\tilde{S}_q^r$. We can sum up the intent-aware representations after r layers as $e_q^u = \left(e_q^{u\,(0)}, e_q^{u\,(1)}, \cdots, e_q^{u\,(r)}\right)$ and $e_q^i = \left(e_q^{i\,(0)}, e_q^{i\,(1)}, \cdots, e_q^{i\,(r)}\right)$ for u and i, respectively. Furthermore, we summarized user/item representations as $e^u = \left(e_1^u, \cdots, e_q^u\right)$, $e^i = \left(e_1^i, \cdots, e_q^i\right)$, respectively.

3.2   Semantic-aware Intent Representation Learning

In this section, we aim to extract aspect embeddings from meta-paths of the rich HIN context. The HIN, which records different types of relationships between users and items, carries diverse semantics and is beneficial to intent representation learning. Specifically, such semantics can be reflected in the meta-path schemes of the given HIN, which is some paths defined composites of different node types with diverse edge types. By characterizing meta-path schemes, the complex relations of the involved nodes can be captured, reflecting higher-level semantics to augment user intent learning. Taking the UMU as an example, the path sequence $\mathrm{U}_{u_1} - \mathrm{M}_{m_1} - \mathrm{U}_{u_2}$ defined under such a meta-path can reflect the behavior similarity of u_1 and u_2, while the social influence of u_2 to u_1 is the important aspect that may affect the intent of u_1. This motivates us to leverage the aspect embeddings modeled from meta-paths as the context to refine the user intent representations.

Formally, given the pre-defined meta-path $\mathbf{p}$, we should firstly generate a series of high-quality path instances $\rho = \{u_1, u_2, \cdots, u_l\}$. Here we resort to *Meta-path Based Random Walks* [2], which is a wildly used path sampling strategy that generates path instances that constitute multiple types of nodes, under a specific meta-path $\mathbf{p}$ to further capture both the semantics and structural correlations between various types of nodes. Then we learn the embeddings of the acquired path instances ρ by a Convolution Neural Network (CNN) [15, 6] parameterized by Θ, then adopt the max-pooling operation to derive the final embedding for a meta-path $\mathbf{p}$ by aggregating the embeddings of L selected path instances:

$$\mathbf{c_p} = \text{max-pooling}\left(\{CNN\left(\{\mathbf{X}_i^\rho\}; \Theta\right)\}_{i=1}^L\right) \tag{9}$$

where $\{\mathbf{X}_i^\rho\}$ means the set of embeddings for L path instances of meta-path $\mathbf{p}$. Each $\mathbf{X}_i^\rho$ illustrates the embedding matrix.

The meta-paths carry important semantic meanings, which can guide the intent learning of users. We propose to extract the semantic embedding from meta path embedding $\mathbf{c_p}$, serving as the context information that waits to be incorporated into the latter semantic-aware intent learning. Specifically, the semantic representation of user u can be derived by the embedding lookup operation as:

$$v_\mathbf{p} = \mathbf{c_p}^\top \cdot \boldsymbol{u} \tag{10}$$

where $\boldsymbol{u} \in \mathbb{R}^{1 \times |\mathcal{U}|}$ is the one-hot encoding of user u. The learned $v_{\mathbf{p}}$ is then serves as the aspect embeddings for all user $u \in \mathcal{U}$ under meta-path $\mathbf{p}$. Therefore, we can extract and generate all the aspect embeddings under different meta-paths.

We then perform the semantic-aware intent learning from intent representation e^u and e^i for users and items in Eq. (3), and aim to incorporate semantics as retained in $v_{\mathbf{p}}$ for users and items to learn the semantic-aware intent representations for the latter recommendation. Towards this end, we design a Factorization Machine (FM) operator to instantiate semantics-aware intent representation $\boldsymbol{h}_{\mathbf{p}}$, which denotes the user intent towards different aspects under meta-path $\mathbf{p}$. Formally, we now have obtained $v_{\mathbf{p}} \in \mathbb{R}^{1 \times d}$ as the semantics-aware representation and the intent-aware representations $e^u = (e^u_1, \cdots, e^u_q) \in \mathbb{R}^{1 \times d}$ for user u. Then $\boldsymbol{h}_{\mathbf{p}}$ can be calculated by a FM module:

$$\boldsymbol{h}_{\mathbf{p}} = e^u \odot v_{\mathbf{p}} \tag{11}$$

where $\odot$ denotes the element-wise product.

Lastly, we perform optimization for model parameters. In detail, the semantics-aware intent representation $\boldsymbol{h}_{\mathbf{p}}$ can be incorporated into recommendation models as one additional user representation. Formally, we use the collaborative filtering to calculate the prediction score $\hat{\boldsymbol{y}}_{ui}$ given user and item ID representations as follows:

$$\hat{\boldsymbol{y}}_{ui} = \alpha \boldsymbol{u}^\top \boldsymbol{i} + (1 - \alpha) \boldsymbol{h}_{\mathbf{p}}^\top \boldsymbol{i} \tag{12}$$

where $\boldsymbol{u}$ and $\boldsymbol{i}$ are the ID embeddings given by id mapping techniques in Eq.(1), such as Multi-OneHot [17] and α is the coefficient that describes how much each component contributes to the prediction score. After obtaining the final representation for user/item, we optimize the parameters for $\boldsymbol{h}_{\mathbf{p}}$ in Eq.(11) by using Bayesian Personalized Ranking (BPR) loss, which encourages the prediction of an observed to be higher than its unobserved counterparts user:

$$\mathcal{L}_{\mathrm{BPR}} = \sum_{u,i,j \in \mathcal{D}} -\ln \sigma \left(\hat{\boldsymbol{y}}_{ui} - \hat{\boldsymbol{y}}_{uj} \right) + \lambda \|E\|_2^2 \tag{13}$$

where $\mathcal{D} = \{(u,i) : u \in U, i \in I, j \in I\}$ is the training set and E is the embedding matrix of all users and items.

3.3 Monte Carlo Edge-drop for Explainability

To further explain the recommendation we propose a novel strategy namely Monte Carlo edge-drop, which aims to provide explainable semantics for the recommendation. By optimizing Eq. (13), we finally produce our prediction model denoted as $f(\cdot)$. We further conduct an inference with a HIN with an edge b removed from meta-path $\mathbf{p}$, i.e., removing the influence of attribute b, which generates the prediction as $\hat{\boldsymbol{y}}^s_{ui}$. Thereafter, we define a criterion, which denotes the absolute error variation between $\hat{\boldsymbol{y}}^s_{ui}$ and the original prediction $\hat{\boldsymbol{y}}_{ui}$, to determine the importance of attribute b. If the variation is greater than a threshold δ, we then claim this aspect is influential since it has a significant impact on the prediction.

4   Experiments

In order to thoroughly evaluate and analyze the proposed methodology, we conducted extensive experiments to answer the following research questions:

- (**RQ1**) How does our method compare with other state-of-the-art models?
- (**RQ2**) How does the threshold δ in Monte Carlo edge-drop strategy improve Top-K recommendation?
- (**RQ3**) How does our method explain users' aspects and provide semantic information for the recommendation?

4.1   Settings

We conduct extensive experiments on two publicly available datasets: *Walmart Recruit*[5], and *Douban Book*[6]. *Walmart Recruit* contains historical retail data from 2011 to 2013 as HIN context includes price, discount, user, gender, category type and city and has been widely used for recommendation related research [19]. The ratings of *Walmart Recruit* are the user's rating number of transactions. *Douban Book* includes rich HIN information such as 3 attributes for the user and 4 attributes for the book. The ratings of *Douban Book* are the user's rating number of books. For both two datasets, we binarize the feedback data (i.e., ratings) by interpreting ratings of 5 or higher as positive feedback (i.e., $r = 1$) or lower as negative feedback (i.e., $r = 0$). Moreover, we use negative sampling to randomly sample unobserved items and pair them with the user as negative instances. The statistics detail are summarized in Table 1.

All experiments are conducted on a Linux server with RTX3070 GPU. We adopt three popular metrics including Recall@K, NDCG@K, and Precision@K, where K is set as 1, 10, 20 and 40 in Table 2. Both two datasets are split as a proportion of 80%/10%/10%, train/test/validate set, respectively. A grid search is used to find the best parameter settings. The embedding size is initialized with Xavier and searched in {16,32,64,128}, and learning rate is in {0.001,0.01,0.05,0.1}. The maximum epoch is set as 1000 with an early stopping strategy. Default hyperparameters of SeDLR are: embedding size 128, disentangled layer iteration number n=3, latent intent number q=4, learning rate 0.01. We compare our proposed SeDLR with three kinds of state-of-the-art recommendation methods: (1) HIN-based methods including IF-BPR [24] and MCRec [8]; (2) Graph-based entangled methods including NeuMF [7] and NGCF [20]; (3) Disentangled-based methods including DGCF [21] and M-VAE [16][7].

4.2   RQ1 Performance Comparison

To understand the performance of SeDLR, we adopt deep comparison with multiple state-of-the-art models on Top-K recommendations. The overall statistical

---

[5] https://www.kaggle.com/c/walmart-recruiting-store-sales-forecasting
[6] https://github.com/librahu/HIN-Datasets-for-Recommendation-and-Network-Embedding/tree/master/Douban%20Book
[7] Refer to related work for more details of baselines.

Table 1. Statistic details: density is $\#Connections/(\#Users \cdot \#Items)$, relation is connection number and Avg.Degree of A is $\#Relation/\#A$.

Dataset (Density)	Node	Relation A-B	Avg.Degree of A/B
Walmart Recruit (0.11%)	#User(U): 5,647 #Gender(G): 2 #City(C): 10 Transaction(T): 20,878 #Category Type(CT): 5 #Amount(A): 5,764	#U-G: 5,645 #U-C: 5,645 #U-T: 23,053 #U-U: 0 #T-A: 23,053 #T-CT: 23,053	#U/G: 1/2822.5 #U/C: 1/564.5 U/T: 4.1/1.1 U/U: 0/0 #T/A: 1.1/4.0 #T/CT: 1.1/4610.6
Douban Book (0.27%)	#User(U): 13,024 #Book(Bo): 22,347 #Group(Gr): 2,936 #Author(Au): 10,805 #Publisher(P): 1,815 #Year(Y): 64	#U-Bo: 792,062 #U-U: 169,150 #U-Gr: 1,189,271 #Bo-Au: 21,907 #Bo-P: 21,773 #Bo-Y: 21,192	#U/Bo: 60.8/35.4 #U/U: 13.0/13.0 #U/Gr: 91.3/405.1 #Bo/Au: 1.0/2.0 #Bo/P: 1.0/12.0 #Bo/Y: 1.0/331.1

outcomes can be found in Table 2. On both two datasets, our SeDLR consistently outperforms all other approaches. Especially, SeDLR improves over the strongest baselines at NDCG@20 by 27.7% and 15.2% on *Walmart Recruit*, and *Douban Book*, respectively. Specifically, most improvements are more than 10%, which validates the Monte Carlo edge-drop has critical effects on improving recommendation performance. Additionally, disentangled methods achieved better results than the other two methods in most cases, which justifies the disentangled representation has a better performance by separating intents, therefore our SeDLR adopts it in our method.

Table 2. Overall Performance Comparison: the best results are marked as bold, strongest baselines are marked with underline.

Datasets	Metrics	NeuMF	NGCF	DGCF	M-VAE	IF-BPR	MCRec	SeDLR	Improv.
Walmart Recruit	Recall@1	0.0376	0.0299	<u>0.0421</u>	0.0391	0.0385	0.0381	**0.0476**	13.1%
	Recall@10	0.0401	0.0387	0.0447	<u>0.0472</u>	0.0419	0.0437	**0.0512**	8.5%
	Recall@20	0.0451	0.0430	<u>0.0516</u>	0.0509	0.0479	0.0448	**0.0552**	7.0%
	Recall@40	0.0612	0.0582	0.0572	0.0519	0.0556	<u>0.0622</u>	**0.0672**	8.0%
	Precision@1	0.0301	0.0315	<u>0.0357</u>	0.0322	0.0316	0.0351	**0.0417**	16.8%
	Precision@10	0.0457	0.0385	<u>0.0477</u>	0.0369	0.0399	0.0426	**0.0516**	8.2%
	Precision@20	<u>0.0528</u>	0.0497	0.0519	0.0489	0.0462	0.0512	**0.0556**	5.3%
	Precision@40	0.0609	0.0599	<u>0.0712</u>	0.0603	0.0591	0.0621	**0.0776**	9.0%
	NDCG@1	0.0201	0.0315	<u>0.0362</u>	0.0288	0.0291	0.0343	**0.0415**	14.6%
	NDCG@10	0.0341	0.0392	<u>0.0448</u>	0.0429	0.0409	0.0422	**0.0512**	14.3%
	NDCG@20	0.0396	0.0499	<u>0.0513</u>	0.0489	0.0502	0.0511	**0.0591**	15.2%
	NDCG@40	0.0670	0.0689	0.0711	0.0676	0.0709	<u>0.0712</u>	**0.0823**	15.6%
Douban Book	Recall@1	0.0267	0.0205	<u>0.0333</u>	0.0301	0.0329	0.0324	**0.0387**	16.2%
	Recall@10	0.0311	0.0377	<u>0.0411</u>	0.0339	0.0362	0.0401	**0.0458**	11.4%
	Recall@20	0.0339	0.0252	0.0431	0.0309	0.0396	<u>0.0478</u>	**0.0515**	7.7%
	Recall@40	0.0641	0.0707	<u>0.0749</u>	0.0691	0.0628	0.0481	**0.0801**	6.9%
	Precision@1	0.0302	0.0344	<u>0.0351</u>	0.0325	0.0281	0.0294	**0.0401**	14.2%
	Precision@10	0.0391	0.0402	<u>0.0415</u>	0.0378	0.0356	0.0352	**0.0476**	14.7%
	Precision@20	0.0420	0.0495	<u>0.0538</u>	0.0322	0.0376	0.0309	**0.0541**	0.6%
	Precision@40	0.0599	0.0618	<u>0.0725</u>	0.0425	0.0564	0.0468	**0.0745**	2.8%
	NDCG@1	0.0301	0.0295	0.0327	<u>0.0341</u>	0.0205	0.0202	**0.0395**	15.8%
	NDCG@10	0.0356	0.0441	<u>0.0457</u>	0.0401	0.0398	0.0268	**0.0552**	20.8%
	NDCG@20	0.0391	0.0301	<u>0.0502</u>	0.0425	0.0463	0.0294	**0.0641**	27.7%
	NDCG@40	<u>0.0682</u>	0.0691	0.0663	0.0645	0.0601	0.0507	**0.0813**	19.2%

4.3   RQ2 Aspect Threshold Influence

We conduct extensive experiments to explore the influence of aspect threshold δ in Monte Carlo edge-drop strategy for three popular metrics Recall@K, NDCG@K, and Precision@K on the recommendation. The empirical results can be found in Figure 3. Through the comparison, we observed the highest accuracy existing in the δ value 0.6 for all three metrics on both two datasets with K@20 and K@40. Then the accuracy drops dramatically later, which is reasonable since the Monte Carlo edge-drop start to filter aspects from 0 and leads to improvement. But the accuracy has decreased when dropped too many aspects, which is a lack of inputs. Accordingly, we summarized with 0.6 is the best aspect threshold on HIN-based disentangled network recommendation.
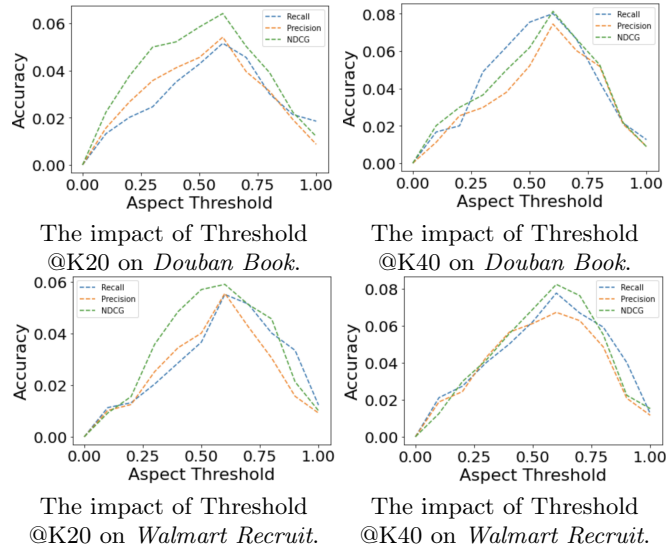

The impact of Threshold @K20 on *Douban Book*.

The impact of Threshold @K40 on *Douban Book*.

The impact of Threshold @K20 on *Walmart Recruit*.

The impact of Threshold @K40 on *Walmart Recruit*.

Fig. 3. The influence of aspect threshold δ in Monte Carlo edge-drop strategy on Top-K recommendation evaluated by Recall@K, NDCG@K and Precision@K.

4.4   RQ3 Model Explainability and Visualization

We visualize two case studies include two users and one item from *Walmart Recruit* to gain a deeper understanding of SeDLR's explainability in Figure 4. By jointly analyzing interaction and aspect, we find the aspect differ across each interaction, which is reflected by score values. For instance, user 268136 only keeps male aspect scoring 0.78, and man store aspects scoring 0.62, under threshold δ value 0.6. It can provide semantics meaning that a male user is likely to interact with an item laptop at the man store. These results show SeDLR not only can effectively untangle users' intents but also add semantic supplements for learned intents.
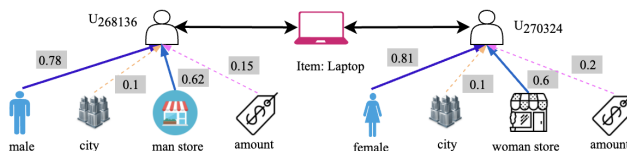
Fig. 4. A visualization of two case studies from *Walmart Recruit*, the bold lines represent retained aspects after Monte Carlo edge-drop strategy with threshold δ value 0.6.

5   Conclusion and Future Work

This paper introduces a novel HIN-based disentangled learning method for Top-K recommendation, namely SeDLR. With the disentangled learning augmented by the HIN, our method is capable of empowering the capability of the recommendation model addressing the bias in historical user interactions. In addition, we resort to Monte Carlo edge-drop strategy to provide the semantic explanations for the recommendation in the real-world datasets. In future work, we will explore the deeper fine-grained level for the item, which is another promising direction.

6   Acknowledgment

References

1. Berg, R.v.d., Kipf, T.N., Welling, M.: Graph convolutional matrix completion. arXiv preprint arXiv:1706.02263 (2017)
2. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: Scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining. pp. 135–144 (2017)
3. Duong, T.D., Li, Q., Xu, G.: Stochastic intervention for causal effect estimation. In: 2021 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2021)
4. Duong, T.D., Li, Q., Xu, G.: Stochastic intervention for causal effect estimation. arXiv preprint arXiv:2105.12898 (2021)
5. Duong, T.D., Li, Q., Xu, G.: Stochastic intervention for causal inference via reinforcement learning. arXiv preprint arXiv:2105.13514 (2021)
6. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: 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. pp. 639–648 (2020)
7. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of the 26th international conference on world wide web. pp. 173–182 (2017)

8. Hu, B., Shi, C., Zhao, W.X., Yu, P.S.: Leveraging meta-path based context for top-n recommendation with a neural co-attention model. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 1531–1540 (2018)

9. Li, Q., Duong, T.D., Wang, Z., Liu, S., Wang, D., Xu, G.: Causal-aware generative imputation for automated underwriting. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management. pp. 3916–3924 (2021)

10. Li, Q., Niu, W., Li, G., Cao, Y., Tan, J., Guo, L.: Lingo: linearized grassmannian optimization for nuclear norm minimization. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. pp. 801–809 (2015)

11. Li, Q., Wang, X., Xu, G.: Be causal: De-biasing social network confounding in recommendation. arXiv preprint arXiv:2105.07775 (2021)

12. Li, Q., Wang, Z., Li, G., Pang, J., Xu, G.: Hilbert sinkhorn divergence for optimal transport. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3835–3844 (2021)

13. Li, Q., Wang, Z., Liu, S., Li, G., Xu, G.: Causal optimal transport for treatment effect estimation. IEEE Transactions on Neural Networks and Learning Systems (2021)

14. Li, Q., Wang, Z., Liu, S., Li, G., Xu, G.: Deep treatment-adaptive network for causal inference. arXiv preprint arXiv:2112.13502 (2021)

15. Lo, S.C.B., Chan, H.P., Lin, J.S., Li, H., Freedman, M.T., Mun, S.K.: Artificial convolution neural network for medical image pattern recognition. Neural networks **8**(7-8), 1201–1214 (1995)

16. Ma, J., Zhou, C., Cui, P., Yang, H., Zhu, W.: Learning disentangled representations for recommendation. arXiv preprint arXiv:1910.14238 (2019)

17. Rodríguez, P., Bautista, M.A., Gonzalez, J., Escalera, S.: Beyond one-hot encoding: Lower dimensional target embedding. Image and Vision Computing **75**, 21–31 (2018)

18. Shi, C., Hu, B., Zhao, W.X., Philip, S.Y.: Heterogeneous information network embedding for recommendation. IEEE Transactions on Knowledge and Data Engineering **31**(2), 357–370 (2018)

19. Taghizadeh, E.: Utilizing artificial neural networks to predict demand for weather-sensitive products at retail stores. arXiv preprint arXiv:1711.08325 (2017)

20. Wang, X., He, X., Wang, M., Feng, F., Chua, T.S.: Neural graph collaborative filtering. In: Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval. pp. 165–174 (2019)

21. Wang, X., Jin, H., Zhang, A., He, X., Xu, T., Chua, T.S.: Disentangled graph collaborative filtering. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1001–1010 (2020)

22. Xu, G., Duong, T.D., Li, Q., Liu, S., Wang, X.: Causality learning: A new perspective for interpretable machine learning. arXiv preprint arXiv:2006.16789 (2020)

23. Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W.L., Leskovec, J.: Graph convolutional neural networks for web-scale recommender systems. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 974–983 (2018)

24. Yu, J., Gao, M., Li, J., Yin, H., Liu, H.: Adaptive implicit friends identification over heterogeneous network for social recommendation. In: Proceedings of the 27th

ACM international conference on information and knowledge management. pp. 357–366 (2018)

25. Zheng, Y., Gao, C., Li, X., He, X., Li, Y., Jin, D.: Disentangling user interest and conformity for recommendation with causal embedding. In: Proceedings of the Web Conference 2021. pp. 2980–2991 (2021)