Causal Disentanglement for Semantics-Aware Intent Learning in Recommendation

Xiangmeng Wang*, Qian Li* †, Dianer Yu, Peng Cui, *Member, IEEE*, Zhichao Wang, Guandong Xu†, *Member, IEEE*

Abstract—Traditional recommendation models trained on observational interaction data have generated large impacts in a wide range of applications, it faces bias problems that cover users' true intent and thus deteriorate the recommendation effectiveness. Existing methods tracks this problem as eliminating bias for the robust recommendation, e.g., by re-weighting training samples or learning disentangled representation. The disentangled representation methods as the state-of-the-art eliminate bias through revealing cause-effect of the bias generation. However, how to design the semantics-aware and unbiased representation for users true intents is largely unexplored. To bridge the gap, we are the first to propose an unbiased and semantics-aware disentanglement learning called **CaDSI** (**Ca**usal **D**isentanglement for **S**emantics-Aware **I**ntent Learning) from a causal perspective. Particularly, CaDSI explicitly models the causal relations underlying recommendation task, and thus produces semantics-aware representations via disentangling users true intents aware of specific item context. Moreover, the causal intervention mechanism is designed to eliminate confounding bias stemmed from context information, which further to align the semantics-aware representation with users true intent. Extensive experiments and case studies both validate the robustness and interpretability of our proposed model.

Index Terms—Causal Disentanglement Learning, Semantics-aware Representation, Causal Intervention.

1 Introduction

Recommender system (RS) has become a panacea for any scenario requiring personalized recommendations, to help users discover users' interested products from overwhelming alternatives. Early works mainly adopt collaborative filtering (CF) methods [1], [2] to model user preference on items, based on historical user-item interactions (e.g., ratings, clicks). However, such user-item interaction usually exhibits bias that is entangled with users' real interests, ignoring degrading the recommendation performance ultimately. For instance, in movie recommendation, users are more likely to watch movies that are watched by many people, which however is due to users' conformity to other people, rather than stemming from users' real interests [3], [4]. Therefore, it is essential to capture users' pure interests that are independent of the bias and thus can be leveraged to build high-quality recommender models.

Most existing works on bias-aware recommendation can be attributed into two categories. The first category adopts a re-weighting strategies on the observed interaction samples, with the aim of imitating the scenario that samples are evenly distributed without bias [5], [6], [7], [8]. One major limitation of these methods is that they merely mitigate bias at the data level, but fail to answer the fundamental question: what are the root causes for bias amplification. Another category of approaches that aim to disentangle user true-interest via inspecting cause-effect of the bias generation for the robust recommendation, have recently gained much attention [4], [9], [10]. Particularly, these works usually design a specific causal graph attributing the bias to a confounder. For example, social network is in fact a confounder for exposure bias, since it influences both users' choice of movie watching and their ratings [9]. Apparently, the confounder introduces pseudo-intents, the ignorance of which definitely misguides the learning of user's true intent. A widely used solution of these works is to learn the user representation that is forced to be independent of the confounder, with the aim of uncovering the user's true intents for final downstream recommendations. Particularly, these works design a regularizer for the independence constraint via statistical measures like L_1-inv, L_2-inv [11] and distance correlation [8]. By explicitly disentangling the cause from a confounder, the representation uncovering user true intent can be learned for final recommendations.

Although promising improvements have been observed, existing approaches on disentangling user intent from bias still suffer two limitations: First, most of them merely focus on user-item relationships, however the interaction data faces sparse issue in practical [12], [13], [14], leading to the

- *X. Wang, D. Yu and G. Xu are with Data Science and Machine Intelligence Lab, Faculty of Engineering and Information Technology, University of Technology Sydney, New South Wales, Australia. E-mail: {Guandong.Xu}@uts.edu.au*
- *Q. Li is with the School of Electrical Engineering, Computing and Mathematical Sciences, Curtin University, Perth, Australia. E-mail: qli@curtin.edu.au.*
- *Z. Wang is with School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, Australia*
- *P. Cui is with the Department of Computer Science and Technology in Tsinghua University, Beijing 100084, China.*

** Both authors contributed equally to this research.*
†Corresponding author.

This work is partially supported by the Australian Research Council (ARC) under Grant No. DP200101374, LP170100891, DP220103717 and LE220100078.

difficulty of learning effective user or item representations. Moreover, existing disentangled learning methods for user intents merely treat one user-item interaction record as an independent instance and neglect its rich context information. We claim that the rich context information in the form of heterogeneous information can help to disentangle and interpret semantics-aware intents of users for the robust recommendation. For instance, higher-order graph structure like a meta path *User-Movie-Actor-Movie-User* encodes the semantics interpretation of "movies starring the same actor rated by the users". In other words, without considering rich context information, disentangled learning fails in offering fine-grained interpretability in terms of item attributes for recommendation.

Therefore, in this work, we propose to enhance user intent disentangled learning with heterogeneous information, which however is not trivial. The heterogeneous information is complicated and consists of various types of data, e.g., item attributes. The complexity in heterogeneous information, e.g., the fact that items grouped by attributes (e.g., *brand*) are frequently with skewed distributions, can bias the user preference and prediction score. The skewed distribution is attributed to missing values of aspects, i.e., the number of non-missing aspects is not evenly distributed in observational dataset. An empirical study conducted on `Douban Movie` dataset can validate this claim by Figure 1: unobserved *Director* aspect accounts for 19.7% of items compared to *Actor* accounting for 7.6%. That is, the skewed distribution of context aspect can easily bias the prediction model towards the majority group, even though their items have the same matching level (see example in Figure 1).

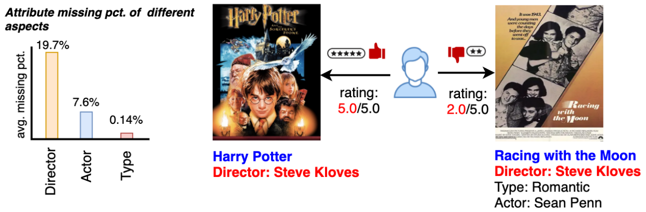


Fig. 1. An example of bias: movie `HP` contains only one aspect *Director*, however, *Actor* and *Type* are missing. The high rating of user on movie `HP` trains prediction model towards the user's preference on the director *Steve Kloves*. In contrast, we observed that movie `RWM` with the same director received very low ratings from the same user.

In this work, we attempt to tackle the challenge from a novel causal perspective, with the aim of developing a unbiased and interpretable disentangled approach on heterogeneous information, named **CaDSI** (**Ca**usal **D**isentanglement for **S**emantics-Aware **I**ntent Learning). To make users' intents semantics-aware, we propose a pre-trained model as a first stage to leverage multiple item facets of heterogeneous information. As a second stage, besides considering the items directly interacted with the user, the higher-order interacted items via meta paths are exploited to disentangle user intents in a robust manner. Finally, the pre-trained model together with disentangled learning is subsequently fine-tuned by the causal intervention. Thanks to the development of causal inference, the second module is designed to adopt causal intervention mechanism to eliminate the bias introduced by heterogeneous information. With these two stages, our method can guide the unbiased and semantics-aware representations disentangling user intents for the robust recommendation. Overall, the key contributions of this work are fourfold:

- Fundamentally different from previous works, CaDSI is the first method that can disentangle the unbiased user's intents from a causal perspective, in the meanwhile endow each user intent with specific semantics under the disentanglement learning task.
- We design a novel causal graph for the qualitative analysis of causal relationships in recommendation, based on which a pre-trained model is designed. With heterogeneous information, the pre-trained model can semantically account for the item context influence towards the user intent.
- To eliminate confounding bias stemmed from heterogeneous information, we perform the causal intervention on user representation and refine the pre-trained model for unbiased user intent learning.
- We conduct extensive experiments to show that our CaDSI method outperforms state-of-the-art methods. The interpretability of our CaDSI is also validated by our empirical study.

2 Preliminary and Problem Formulation

In this section, we will first present causal relations underlying the data generation mechanism of recommendation with access to the heterogeneous information. Following this, we prove the existence of confounder in the heterogeneous information and discuss the consequences for ignoring the context bias bought by the confounder. We then introduce the causal intervention and discuss how intervention via back-door adjustment can control the context bias from a causal perspective. Finally, we introduce the important concepts and notations used in our approach and give the formal definition of our problem to be solved.

2.1 Problem definition

We formulate our task as disentangling and interpreting users' intent based on Heterogeneous Information Network (HIN). The important concepts of HIN and the formal definition of our problem are given as follows.

Definition 1 (Heterogeneous Information Network)**.** *A Heterogeneous Information Network (HIN) is denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with a node type mapping function: $\phi : \mathcal{V} \rightarrow \mathcal{A}$ and an edge type mapping function: $\psi : \mathcal{E} \rightarrow \mathcal{R}$, where $\mathcal{A}$ and $\mathcal{R}$ are the node type set and edge type set of $\mathcal{G}$, respectively. Each node $v \in \mathcal{V}$ and edge $e \in \mathcal{E}$ in a HIN belongs to one particular type in node/edge type sets $\mathcal{V}/\mathcal{E}$ with $\phi(v) \in \mathcal{A}$ and $\psi(e) \in \mathcal{R}$, where $|\mathcal{A}| + |\mathcal{R}| > 2$.*

Definition 2 (Meta Path)**.** *Meta path $\mathbf{p}$ is a path defined on the network schema $T_{\mathcal{G}} = (\mathcal{A}, \mathcal{R})$, and is denoted in the form of*

$$\mathbf{p} \triangleq (\mathcal{A}_1 \overset{\mathcal{R}_1}{\rightarrow} \mathcal{A}_2 \overset{\mathcal{R}_2}{\rightarrow} \ldots \overset{\mathcal{R}_l}{\rightarrow} \mathcal{A}_{o+1})$$

which defines a composite $\mathcal{R} = \mathcal{R}_1\mathcal{R}_2...\mathcal{R}_o$ between type $\mathcal{A}_1$ and $\mathcal{A}_{o+1}$. For simplicity, we use node type names to denote the meta path if no multiple relations exist between type pairs, as $\mathbf{p} = (\mathcal{A}_1\mathcal{A}_2...\mathcal{A}_{o+1})$. Commonly, a HIN contains multiple meta paths, the meta path set is defined as $\mathcal{P}$ where each $\mathbf{p} \in \mathcal{P}$.

Based on these important concepts, we collect the key notations in Table 1 and formulate the problem to be solved as follows.

Definition 3 (Problem Definition)**.** *Given user and item sets, we define an interaction matrix $\boldsymbol{y} \in \mathbb{R}^{m\times n}$ where entry $\boldsymbol{y}_{ui} = 1$ indicates a user u in user set interacts with an item i in item set, otherwise $\boldsymbol{y}_{ui} = 0$. We also have additional contextual information about users and items, e.g., social relationships between users or item brands and categories, absorbing in $\mathcal{G}$ of Definition 1. Thus, we aim to learn the prediction function P parameterized by Θ, such that $\hat{\boldsymbol{y}}_{ui} = P(u, i|\boldsymbol{y}, \mathcal{G}; \Theta)$, where $\hat{\boldsymbol{y}}_{ui}$ denotes the probability that user u will engage with item i conditional on the given $\boldsymbol{y}$ and $\mathcal{G}$.*

2.2 A Causal View on Recommendation

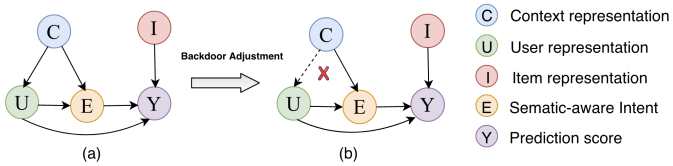


Fig. 2. Causal disentanglement model for recommendation. We apply backdoor adjustment to remove the effect of confounder C for U, as indicated by the red cross.

2.2.1 Structural Causal Model

To illustrate the recommendation data generation mechanism seriously and soberly, we consider the structural causal model (SCM) [15] based on causality to reveal the true causal relations in recommendation. The basic idea of our proposed approach is to disentangle and interpret users' intent based on heterogeneous information. From a causal perspective, Figure 2 (a) demonstrates the illustrative causal graph that offers an interpretable representation for disentangled recommender system, which consists of four variables including $\{U, C, E, Y\}$. In particular, as a directed acyclic graph, it can describe the generation mechanism of recommendation results and guide the design of recommendation methods. In the following, we explain the rationality of this causal graph at a higher-level.

- C as a confounder is the item aspects (e.g., movie genre) acquired from the heterogeneous information network. The representation of C can be learned by a pre-trained model of context information, which retains semantics information of item aspects.
- U denotes user representation which essentially reveals k user intents. U is presented in the form of k chunked intent representation, where each chunk of representation reveals a piece of user intent, such as the user's special taste towards items' brand.
- I is item representations and each I denotes the embedding of one item attribute (e.g. *Genre*).
- E is the semantics-aware intent representation generated by the context information from C and the user representation U. E retains the information of the user intent towards different item aspects.
- $Y \in [0, 1]$ is the recommendation probability for the user-item pair.

The directed edge represents the causal relation between two variables, in particular, the rationality of causal relations can be explained as follows.

- $C \rightarrow U$: The prior knowledge C of item aspects affect user representation U, which is reflected by the fact that users prefer the items who have particular attributes (e.g., brand).
- $(C, U) \rightarrow E$: Item context C and user representation U consist of the semantics-aware user intent representation.
- $I \rightarrow Y$: item representation by I affects the recommendation probability Y.
- $U \rightarrow Y$: user's preference represented by U affects the recommendation probability Y.
- $U \rightarrow E \rightarrow Y$: the recommendation probability of item could be high if the user shows interest in the context of the item, e.g., the item type rather than the item. For example, items whose type is "lipstick" are more likely to be purchased by the user u whose gender is female.

2.2.2 Adjusting Confounding Bias via Intervention

From this causal graph, the semantic knowledge C is a confounder between user representation U and recommendation outcome Y, since C is the common cause of U and Y by definitions in causal theory. The presence of confounder C leads to the spurious correlation between U and Y if we ignore to account its causal effect into modeling, which is equal to the estimation of $P(Y \mid U)$. In semantic knowledge-aware recommendation, the confounder C (i.e., semantic knowledge) makes recommendation probability $P(Y \mid U)$ biased towards items that have dominant item attributes. For example, as illustrated in Figure 1, we expect that the rating prediction of `RWM` is caused by both of the three item attributes, but not only the dominant item attribute *director* which has a majority attribute popularity (i.e., the majority group). In the language of causal inference, the conventional correlation $P(Y \mid U)$ fails to capture the true causality between U and Y, because the prediction likelihood of Y is conditional on not only U, but also the spurious correlation via (1) $C \rightarrow U \rightarrow Y$, i.e., prior knowledge C determines the prediction likelihood through user representation U. For example, the undesirable and low-quality items in the specific attribute group will not attract users' intent, degrading recommendation accuracy. (2) $C \rightarrow E \rightarrow Y$. i.e., the semantic-ware user intent representation E derived from C affects the prediction Y. Once users' future interest in item attribute groups changes (i.e., user interest drift), the recommendations will be biased.

To pursue the true causality between U and Y, we should propose a causal intervention method $P(Y \mid do(U))$ to remove the confounding bias from C. The $do(\cdot)$ operation [15] is to forcibly and externally assign a certain value to the variable U, which can be intuitively seen as

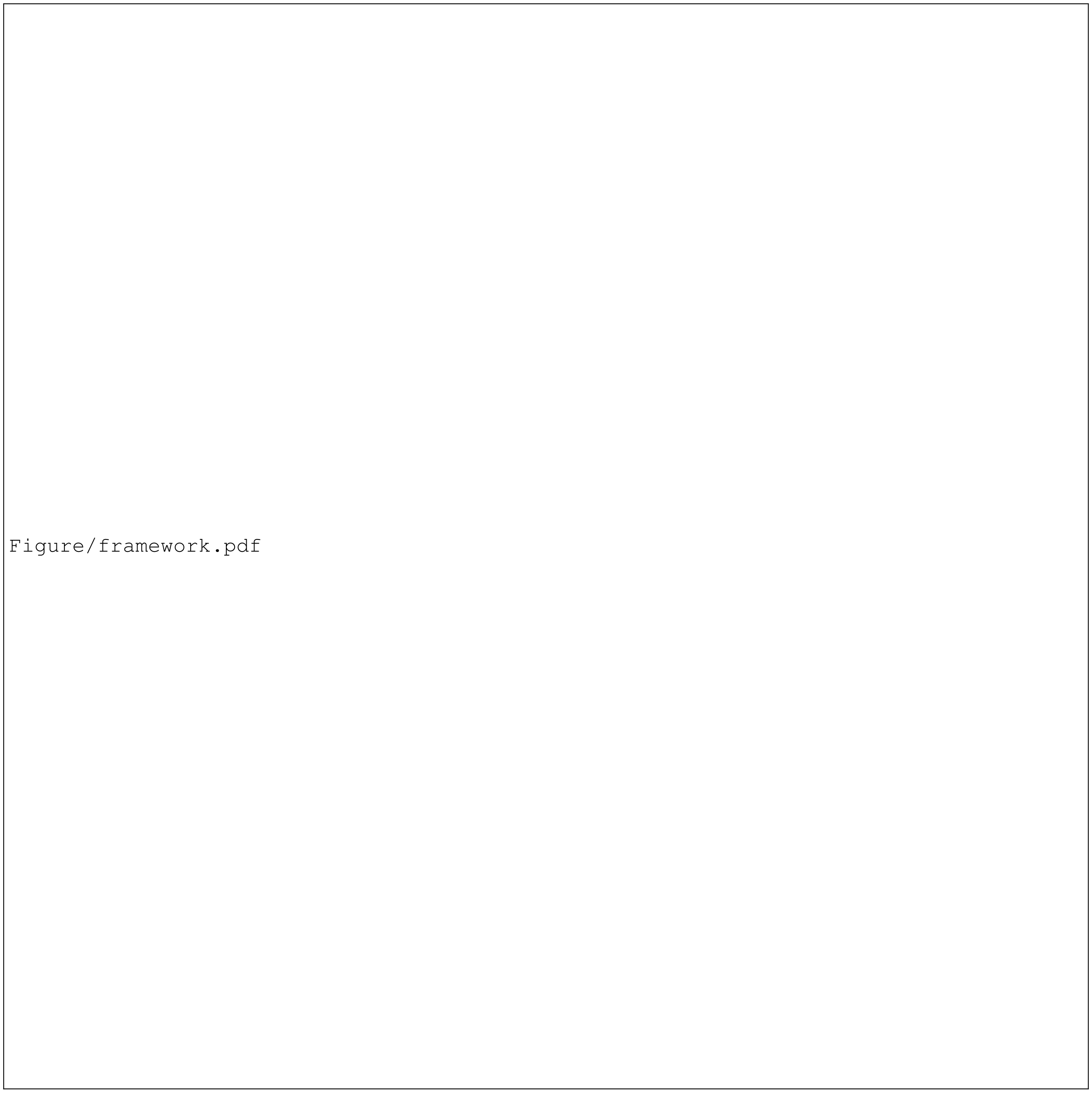


Fig. 3. Overview of the proposed CaDSI. CaDSI takes HIN as the input, and passes the causal disentanglement model for learning intent-aware representations (*cf.* Section 3.3); then use the causal intervention (*cf.* Section 3.4) for controlling the counfounding bias.

removing the edge $C \rightarrow U$ and blocking the effect of C on U (as shown in Figure 2 (b)). As the result, the prediction likelihood can be independent of its causes, so as to generate better recommendation performance that is free from the confounding bias.

3 Our method

In this section, we first introduce motivation and the overall architecture of the proposed model, which includes *Causal Disentanglement Model* and *Causal Intervention* as shown in Figure 3. We then present the details of each component and how they are applied to top-N recommendation.

3.1 Pre-trained Model for Learning Context Information

The pre-trained model for learning context representation C is a key component in the causal disentanglement model, which aims to leverage side-information in the given HIN and construct expressive representations for users, items and aspects directly. Specifically, Given a HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and its corresponding meta paths set $\mathcal{P}$, we aim to learn semantics-aware representation (a.k.a., embedding) $\boldsymbol{c_u}$ for each user node (i.e, *User* type) that represents the semantic-aware embedding for user $u \in \mathcal{V}$, $\boldsymbol{c_i}$ for each item node (i.e, *Item* type) that represents the semantic-aware embedding for item $i \in \mathcal{V}$ and $\boldsymbol{c_a}$ for each aspect node $a \in \mathcal{V}$ that represents context information representation of a specific

TABLE 1
Key notations and descriptions.

Notation	Description
$\mathcal{G}$	Heterogeneous Information Network (HIN)
$\mathcal{V}$	node set of HIN
$\mathcal{A}$	node type set of HIN
$\mathcal{P}$	meta paths set of HIN
$\mathbf{p}$	a meta path in $\mathcal{P}$
$\boldsymbol{y} \in \mathbb{R}^{m\times n}$	user item interaction matrix
$\hat{\boldsymbol{y}}_{ui}$	predicted interaction likelihood of user u and item i
$\boldsymbol{c_u}$	semantics-aware embedding for user u
$\boldsymbol{c_i}$	semantics-aware embedding for item i
$\boldsymbol{c_a}$	context information embedding for aspect a
k	user intent number
$\boldsymbol{l}$	iteration number of graph disentangling module
L	L-th layer of graph disentangling module
$\mathbf{S}_k(u,i)$	intent score of u and i on intent k
$\boldsymbol{u}^u$	intent-aware embedding for u
$\boldsymbol{i}^i$	embedding for item i

type of aspect a (e.g., *Director*).

Towards this, a *Heterogeneous Skip-Gram with Meta Path Based Random Walks* is designed to output a set of multinomial distributions, while each distribution corresponding to one type of node (i.e, *User*, *Item* and *Aspect* type); the *Meta Path Based Random Walks* is used to generate node sequences that capture the complex semantics reflected in a Heterogeneous Information Network (HIN), while *Heterogeneous Skip-Gram* takes the generated node sequences as inputs and catches the heterogeneous neighborhood of a node for outputting the semantics-aware embeddings. Finally, the semantics-aware embeddings for users, items and aspects are given by aggregating every node representation under different meta paths by an *Embedding Fusion* operation.

3.1.1 Meta Path Based Random Walks

To generate node sequences that are able to capture both the semantics and structural correlations between different types of nodes. The *Meta Path Based Random Walks* [16] is proposed to generate the node sequences traversed by random walkers over a HIN. The basic idea is to put random walkers [17] in a HIN to generate paths that constitute multiple types of nodes. Specifically, given $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R}, \phi, \psi)$, the node sequence $\mathbf{n_p} = \{v_1, \cdots, v_{i+1}\}$ under a specific meta path $\mathbf{p}$ is generated according to the following distribution:

$$P\left(v_{i+1} \mid v_i, \mathbf{p}\right) \\ = \begin{cases} \frac{1}{\left|\mathcal{N}_{v_i}^{(\mathcal{A}_{o+1})}\right|}, & (v_i, v_{i+1}) \in \mathcal{E} \text{ and } \phi(v_{i+1}) = \mathcal{A}_{o+1} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $\mathcal{N}_{v_i}^{(\mathcal{A}_{o+1})}$ is the first-order neighbor set for node v_i whose type is $\mathcal{A}_{o+1}$; v_{i+1} is the $i+1$-th node whose type is $\mathcal{A}_{o+1}$, and v_i is the i-th node in the walk which belongs to type $\mathcal{A}_o$. By regulating $v_i \in \mathcal{A}_o$ while $v_{i+1} \in \mathcal{A}_{o+1}$, the node types sampled by random walkers is conditioned on the pre-defined meta path $\mathbf{p}$.

Following the pre-defined meta paths in Table 3, by performing the *Meta Path Based Random Walks* on each meta path $\mathbf{p} \in \mathcal{P}$, we can obtain node sequences set for all meta paths as $\mathbf{n}^{\mathcal{P}} = \{\mathbf{n}_1, \cdots, \mathbf{n}_{|\mathcal{P}|}\}$. As we care about sematic-aware embeddings for users, items and aspects, we then select meta paths $\mathbf{p}$ starting with *user*, *item* or *Aspect* type and reorganize its corresponding node sequences $\mathbf{n_p}$ into *user* type-specific set $\mathbf{n}(U)$ as $\mathbf{n}(U) = \{\mathbf{n}_1, \cdots, \mathbf{n}_m\}$, *item* type-specific set $\mathbf{n}(I)$ as $\mathbf{n}(I) = \{\mathbf{n}_1, \cdots, \mathbf{n}_n\}$ and aspect type-specific set $\mathbf{n}(A) = \{\mathbf{n}_1, \cdots, \mathbf{n}_h\}$. We then use *Heterogeneous Skip-Gram* to generate semantics-aware embeddings of node sequences in $\mathbf{n}(U)$, $\mathbf{n}(I)$ and $\mathbf{n}(A)$.

3.1.2 Heterogeneous Skip-Gram

Based on node sequences in set $\mathbf{n}(U)$, $\mathbf{n}(I)$ and $\mathbf{n}(A)$, we aim to leverage *Heterogeneous Skip-Gram* [16] to learn node representations $\boldsymbol{c}_u$ and $\boldsymbol{c}_i$ and $\boldsymbol{c}_a$. Note that $\boldsymbol{c}_u$ and $\boldsymbol{c}_i$ and $\boldsymbol{c}_a$ represent semantics-aware user, item and aspect representations of a specific node sequence $\mathbf{n}_i$ in $\mathbf{n}(U)$, $\mathbf{n}(I)$ and $\mathbf{n}(A)$, respectively. For concise purpose, we only present the learning process of the node representation $\boldsymbol{c}_u$, and analogously, we can obtain the representations of $\boldsymbol{c}_i$ and $\boldsymbol{c}_a$.

Specifically, a *Heterogeneous Skip-Gram* is designed to learn node representations by aggregating the heterogeneous neighborhood of the node in node sequence, while optimized through a node type-specific negative sampling [16]. Given each node sequence $\mathbf{n}_i$ in $\mathbf{n}(U)$ generated from Eq. (1), the Skip-Gram model learns the semantics-aware embedding $\boldsymbol{c}_u$ of $\mathbf{n}_i$ by maximizing the probability of having the heterogeneous context $\mathcal{N}_u$ given a node u as follows:

$$\mathcal{L}_\theta = \sum_{u\in\mathcal{V}} \sum_{u_c \in \mathcal{N}_u^{\mathcal{A}_i}} \sum_{\mathcal{A}_i \in \mathcal{A}} \left(\sigma\left({\boldsymbol{c}_u}^T \boldsymbol{c}_{u_c}\right) \prod_{w=1}^{W} \sigma\left({\boldsymbol{c}_u}^T \boldsymbol{c}_w\right); \theta \right) \tag{2}$$

where $\mathcal{N}_u^{\mathcal{A}_i}$ denotes u's neighborhood whose type is $\mathcal{A}_i$, u_c is one node in the neighborhood set $\mathcal{N}_u$ of u, $\boldsymbol{c}_u$ and $\boldsymbol{c}_{u_c}$ are latent vectors that correspond to the target node and context node representations of u and u_c, and $\sigma(x) = 1/1 + \exp(-x)$. W is a parameter that determines the number of negative examples to be drawn per a positive example, $\boldsymbol{c}_w$ is the sampled node's representation within W negative samples and θ is the model parameters of *Heterogeneous Skip-Gram*. Finally, $\boldsymbol{c}_u$ for each node sequence $\mathbf{n}_i$ in $\mathbf{n}(U)$ are estimated by applying gradient descent algorithm [18] with respect to the objective in Eq. (2).

3.1.3 Embedding Fusion

Since we have multiple node sequences in $\mathbf{n}(U)$, $\mathbf{n}(I)$ and $\mathbf{n}(A)$, while each learned representation $\boldsymbol{c}_u$, $\boldsymbol{c}_i$ and $\boldsymbol{c}_a$ is the semantic-aware embedding of each node sequence $\mathbf{n}_i$ in $\mathbf{n}(U)$, $\mathbf{n}(I)$ and $\mathbf{n}(A)$ respectively, we therefore perform embedding fusion to aggregate every representations $\boldsymbol{c}_u$, $\boldsymbol{c}_i$ and $\boldsymbol{c}_a$ into an uniform manner categorized by their node types so as to guide the recommendation task. The reason why we fuse the individual embedding of each node sequence is quite straightforward. Firstly, in recommendation system, the optimization goal is to learn effective representations for users and items. Hence, it requires a principled fusion way to transform node embeddings w.r.t. different meta paths relating *user* type or *item* type into a more suitable form for later recommendation tasks. Secondly, the context information across meta paths starting with an aspect type should be further arranged into an uniform embedding space, representing one piece of semantics meaning, e.g., the aspect of the object been "Director".

The embedding fusion is implemented as a liner combination function defined as follows:

$$\begin{aligned}
\boldsymbol{c}_{\boldsymbol{u}} &\leftarrow \frac{1}{|\boldsymbol{c}_u(U)|} \sum_{j=1}^{|\boldsymbol{c}_u(U)|} \left(\mathbf{M} \cdot \boldsymbol{c}_u^j + b\right) \\
\boldsymbol{c}_{\boldsymbol{i}} &\leftarrow \frac{1}{|\boldsymbol{c}_i(I)|} \sum_{j=1}^{|\boldsymbol{c}_i(I)|} \left(\mathbf{M} \cdot \boldsymbol{c}_i^j + b\right) \\
\boldsymbol{c}_{\boldsymbol{a}} &\leftarrow \frac{1}{|\boldsymbol{c}_a(A)|} \sum_{j=1}^{|\boldsymbol{c}_a(A)|} \left(\mathbf{M} \cdot \boldsymbol{c}_a^j + b\right)
\end{aligned} \tag{3}$$

where $\boldsymbol{c}_u(U)$ is the user node representation set which absorb the node representations of user u and $\boldsymbol{c}_u(U) = \{\boldsymbol{c}_u^1, \cdots, \boldsymbol{c}_u^m\}$. Correspondingly, the item and aspect node representation sets $\boldsymbol{c}_i(I) = \{\boldsymbol{c}_i^1, \cdots, \boldsymbol{c}_i^n\}$ and $\boldsymbol{c}_a(A) = \{\boldsymbol{c}_a^1, \cdots, \boldsymbol{c}_a^h\}$ are established for item i and aspect a. $\mathbf{M}$ is a linear combination transformation matrix [19] and $\boldsymbol{b}$ is the error term. Through Eq. (3), $\boldsymbol{c}_{\boldsymbol{u}}$, $\boldsymbol{c}_{\boldsymbol{i}}$ and $\boldsymbol{c}_{\boldsymbol{a}}$ can be learned as final semantics-aware representations for a user u and an item i and context information representation for aspect a, respectively.

3.2 Disentanglement Learning for User Intent

Inspired by recent achievements on GNNs [12], [13], [20], [21], we propose a GNN-based disentangling module to learn user representations that can essentially reveal k user intents. Specifically, the L-layer disentangling module exploits the high-order connectivities among user-item interaction graph and initializes intent-ware embeddings by separating each user/item embedding into k chunks. Then, an interactive update rule that computes the importance scores of intent-aware user-item interactions is designed to refine intent-aware embeddings, so as to disentangle the holistic interaction graph into k intent-ware sub-graphs. Thereafter, each intent-aware embedding chunk are stacked by embedding propagation in the current layer, serving as the holistic intent-aware embedding $\boldsymbol{u}^u$ and $\boldsymbol{i}^i$ for user u and item i, where each intent-aware embedding $\boldsymbol{u}^u$ and $\boldsymbol{i}^i$ is composed of k independent chunks:

$$\boldsymbol{u}^u = [\boldsymbol{u}_1^u, \cdots, \boldsymbol{u}_k^u], \quad \boldsymbol{i}^i = [\boldsymbol{i}_1^i, \cdots, \boldsymbol{i}_k^i] \tag{4}$$

Each chunked representation $\boldsymbol{u}_k^u \in \mathbb{R}^{\frac{d}{k}}$ and $\boldsymbol{i}_k^i \in \mathbb{R}^{\frac{d}{k}}$is built upon the intent-aware interactions between user u and its preferred items under intent i. We ultimately sum up the intent-aware representations at each intent k of all L layers, the final layer outputs the k-th chunked intent-aware representations $\boldsymbol{u}_k^u$ and $\boldsymbol{i}_k^i$:

$$\boldsymbol{u}_k^u = \boldsymbol{u}_k^{u(1)} + \cdots + \boldsymbol{u}_k^{u(L)}, \quad \boldsymbol{i}_k^i = \boldsymbol{i}_k^{i\,(1)} + \cdots + \boldsymbol{i}_k^{i\,(L)} \tag{5}$$

The detailed operations are given as follows.

3.2.1 Initialization

As ID embedding captures intrinsic characteristics of users, we separate the ID embeddings of user u into k chunks and associate each chunk with a latent intent, serving as the initialization of the intent-aware embeddings $\boldsymbol{u}^u$ in Eq. (4):

$$\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_k] \tag{6}$$

Thereafter, we initialize the importance score $\mathbf{S}_k(u,i)$. The $\mathbf{S}_k(u,i)$ is the importance score of the interaction between u and i with respect to the k-th intent. Such $\mathbf{S}_k(u,i)$ can be seen as the indicator of whether u should interact with i under a specific intent k. Thus, by learn $\mathbf{S}_k(u,i)$ for aspect $k \in \{1, \cdots, k\}$, we can construct an intent-aware sub-graph at intent k. Such that, the embedding propagation can output intent-aware representation at intent k for all users based on the corresponding intent-aware sub-graph. We uniformly initialize importance score $\mathbf{S}_k(u,i)$ as $\mathbf{S}_k(u,i) = \frac{1}{k}$ which presumes the equal contributions of intents at the start of modeling.

3.2.2 Iterative Update Rule

An iterative update rule is then designed to update the importance score $\mathbf{S}_k(u,i)$ of user-item connections under aspect k within $\boldsymbol{l}$ iterations, so as to disentangle interaction sub-graph at intent k to refine each intent-aware embedding chunk $\boldsymbol{x}_k$ in Eq. (6). Note that $\boldsymbol{x}_k \in \boldsymbol{x}$ in Eq. (6) serves as the initialized representation chunk of $\boldsymbol{u}_k^u \in \boldsymbol{u}^u$ in Eq. (4) and is used to memorize the update value during iteration, the final $\boldsymbol{x}_k$ is assigned to each $\boldsymbol{u}_k^u$ as the final intent-aware representation chunk of user u.

In particular, we set $\boldsymbol{l}$ iterations in the interactive update. At each iteration, for the target interaction (u,i), we firstly normalize the score vector $\mathbf{S}_k(u,i) \mid \forall k \in \{1, \cdots, k\}\}$ over all intents into $\tilde{\mathbf{S}}_k$ through a softmax function:

$$\tilde{\mathbf{S}}_k^{\boldsymbol{l}}(u,i) = \frac{\exp\left(\mathbf{S}_k^{\boldsymbol{l}}(u,i)\right)}{\sum_{k'=1}^{k} \exp\left(\mathbf{S}_{k'}^{\boldsymbol{l}}(u,i)\right)} \tag{7}$$

which is capable of illustrating which intents should get more attention to explain each user behavior (u,i). We then perform embedding propagation over individual intent-aware graphs whose representation is denoted by $\boldsymbol{x}_k$, $\boldsymbol{i}_k^i$ and its adjacency matrix is denoted by $\tilde{\mathbf{S}}_k$ in Eq. (7), such that the information of all individual intent-aware graphs are encoded into the learned representations. The weighted sum aggregator is defined as:

$$\boldsymbol{x}_k^{\boldsymbol{l}} = \sum_{i \in \mathcal{N}_u} \frac{\tilde{\mathbf{S}}_k^{\boldsymbol{l}}(u,i)}{\sqrt{D_k^{\boldsymbol{l}}(u) \cdot D_k^{\boldsymbol{l}}(i)}} \cdot \boldsymbol{i}_k^i \tag{8}$$

where $\boldsymbol{x}_k^{\boldsymbol{l}}$ is $\boldsymbol{u}_k^u$'s temporary representation that memorizes information acquired from u's neighbors $\mathcal{N}_u$ at $\boldsymbol{l}$-th iteration, $D_k^{\boldsymbol{l}}(u) = \sum_{i' \in \mathcal{N}_u} \tilde{\mathbf{S}}_k^{\boldsymbol{l}}(u,i')$ and $D_k^{\boldsymbol{l}}(i) = \sum_{u' \in \mathcal{N}_i} \tilde{\mathbf{S}}_k^{\boldsymbol{l}}(u',i)$ are the degrees of user u and item i, respectively.

Then we interactively update the intent-aware graphs. Intuitively, historical items of users driven by the same intent tend to have the similar chunked representations, such goal can be achieved by encouraging users and items among the same intent to have stronger relationships. We hence iteratively update the interaction importance score $\mathbf{S}_k^{\boldsymbol{l}}(u,i)$ in order to strengthen the degree between the centroid u and its neighbor i under intent k, as follows:

$$\mathbf{S}_k^{\boldsymbol{l}+1}(u,i) = \mathbf{S}_k^{\boldsymbol{l}}(u,i) + \boldsymbol{x}_k^{\boldsymbol{l}\top} \tanh\left(\boldsymbol{i}_k^i\right) \tag{9}$$

where $\boldsymbol{x}_k^{\boldsymbol{l}\top} \tanh\left(\boldsymbol{i}_k^i\right)$ considers the affinity between $\boldsymbol{x}_k^{\boldsymbol{l}}$ and $\boldsymbol{i}_k^i$, and $tanh$ [22] is a nonlinear activation function to increase the representation ability of model.

After $\boldsymbol{l}$ iterations, $\boldsymbol{u}^{u(1)} = \boldsymbol{x}^{\boldsymbol{l}(1)}$ for user u is obtained in the current layer, where each chuncked representation $\boldsymbol{u}_k^{u(1)} = \boldsymbol{x}_k^{\boldsymbol{l}\,(1)}$ denotes the chunked embedding of $\boldsymbol{u}^{u(1)}$ on the intent k corresponds to the k-th dimension of Eq. (4).

3.2.3 Layer Combination

To explore high-order connectivity between users and items, we recursively formulate the representation of L-th layer as:

$$\boldsymbol{u}_k^{u(L)} = g\left(\boldsymbol{x}_k^{\boldsymbol{l}\,(L-1)}, \left\{\boldsymbol{i}_k^{i\,(L-1)} \mid i \in \mathcal{N}_u\right\}\right) \tag{10}$$

where $\boldsymbol{u}_k^{u(L)}$ and $\boldsymbol{i}_k^{i\,(L)}$ are the representations of user u and item i on the k-th intent at layer L, $\boldsymbol{x}_k^{\boldsymbol{l}\,(L-1)}$ is the chunked embedding of k-th intent at $L-1$-th layer of $\boldsymbol{u}_k^{u(L-1)}$. Note that $g(\cdot)$ is a fully connection layer that memorizes the information propagated from the $(L-1)$-order neighbors of u.

Finally, after L layers, we sum up intent-aware representations at different layers as the final representation of the k-th chunk of Eq. (4), as $\boldsymbol{u}_k^u = \boldsymbol{u}_k^{u(1)} + \cdots + \boldsymbol{u}_k^{u(L)}$, where $\boldsymbol{u}_k^u$ donates the intent-aware representation for user u at intent k. Analogously, we can establish the final intent-aware embedding chunk $\boldsymbol{i}_k^i$ follow the definition in Eq. (5).

3.3 Semantics-aware Intent Learning

Having obtained the intent-aware embeddings $\boldsymbol{u}^u$, $\boldsymbol{i}^i$ from *disentanglement learning for User Intent*, our method takes semantics factors $\boldsymbol{c_u}$ and $\boldsymbol{c_i}$ from *Pre-trained Model for Context Information* as the auxiliary input for semantics-aware recommendation. To facilitate the usage of semantics factors, we design an operator to instantiate a semantics-aware intent representation $\boldsymbol{e}$, which denotes the user intent towards different aspects. By learning $\boldsymbol{e}$, the effect of semantics factors towards user intent can be incorporated into the final updated user representation $\boldsymbol{u}^u$, such that $\boldsymbol{u}^u$ can be easily plugged into the backdoor adjustment to alleviate bias. Specifically, a second-order Factorization Machine (FM) [23] module is used to instantiate $\boldsymbol{e}$:

$$\boldsymbol{e} = \sum_{a=1}^{d}\sum_{b=1}^{d} \boldsymbol{u}_a^u \boldsymbol{c}_{\boldsymbol{i}_a} \odot \boldsymbol{c}_{\boldsymbol{u}_b} \boldsymbol{i}_b^i \tag{11}$$

where $\odot$ denotes the element-wise product between each latent vector, such that the learned $\boldsymbol{e}$ captures the interactions between the intent-aware representation $\boldsymbol{u}^u/\boldsymbol{i}^i$ and semantics factors in $\boldsymbol{c_u}/\boldsymbol{c_i}$.

Next, the semantics-aware intent representation $\boldsymbol{e}$ can be incorporated into recommender 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:

$$\hat{\boldsymbol{y}}_{ui} = f(\boldsymbol{u}, \boldsymbol{i}, \boldsymbol{e}) = \delta \boldsymbol{u}^\top \boldsymbol{i} + (1-\delta)\boldsymbol{e}^\top \boldsymbol{i} \tag{12}$$

where $\boldsymbol{u}$ and $\boldsymbol{i}$ are the ID embeddings given by id mapping techniques such as Multi-OneHot [24], and δ is the coefficient that describes how much each component contributes to the prediction score. Then we use the pairwise BPR loss [25] to optimize the model parameters Θ. Specifically, BPR loss encourages the prediction of a user's historical items to be higher than those of unobserved items:

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

where $\boldsymbol{u}$, $\boldsymbol{i}$ and $\boldsymbol{j}$ are the ID embeddings of user u, item i and item j, $O = \{(u,i,j) \mid (u,i) \in O^+, (u,j) \in O^-\}$ denotes the training dataset involving the observed interactions O^+ and unobserved counterparts O^-; $\sigma(\cdot)$ is sigmoid function; λ is the coefficient controlling regularization.

3.4 Causal Intervention for Debiasing

The context information as a confounder tends to introduce bad effect on both user representation and prediction score. To make context information beneficial for semantics-aware intent learning, we resort to causal technique, backdoor adjustment [15] to adjust the representation mechanism in disentangled causal model. By doing this, we aim to produce unbiased user representation and apply the modified one to recommendation task.

3.4.1 Backdoor Adjustment

Note that previous recommendation methods build a predictive model $P(Y \mid U)$ from the passively collected interaction dataset, which neglects the effect of confounder C, thus leading to a spurious correlation between users and items. The spurious correlation is harmful to most users because the items in the majority group are likely to dominate the recommendation list and narrow down the user interests.

According to the theory of backdoor adjustment [15], the target of our method is to remove the bad effect of context information C on user representation U. Instead of $P(Y \mid U)$, we formulate the predictive model as $P(Y|do(U=u))$ to account for the effect of confounder. Based on the graph in Figure. 2, we first need to formulate the causal effect between variables by causal intervention, which is denoted as $do(\cdot)$ operation [15]. The operation $do(U=u)$ is defined to externally assign a certain value to the variable U. Namely, $do(U=u)$ intuitively removes the edge $C \rightarrow U$ so as to block the effect of C on U, making the value of U independent of its causes (*cf.* Figure. 2). By applying do operation, we can estimate the effect of C on Y as

$$P(Y|do(U=u)) - P(Y|do(U=0)) \tag{14}$$

where $P(Y|do(U=0))$ denotes the null intervention, e.g., the baseline compared to $U=u$. In the physical world, $P(Y|do(U=u))$ corresponds to actively manipulating the aspects or attributes in the item.

Our implementation is inspired from two inherent properties of any Heterogeneous Network Embedding method (e.g., metapath2vec++ [16]). First, by passing a meta-path into the pre-trained context model from Eq. (3), we have the context embedding for an aspect, denoted as $\boldsymbol{c_a}$. Aggregating context embeddings for all aspects into the aspect representation set C, we can obtain the unified aspect representation C, where each element of C, i.e., $\boldsymbol{c_a}$, representing one semantic aspect (e.g., "Director") is computed by Eq. (3). As C is no longer correlated with $\boldsymbol{u}^u$ by removing the edge $C \to U$, the causal intervention makes $\boldsymbol{u}^u$ have a fair opportunity to incorporate every context $\boldsymbol{c_a}$ into the prediction of $\hat{\boldsymbol{y}}_{ui}$, subject to a prior $P(C = \boldsymbol{c_a})$. Second, prevailing pre-trained models use a specific task (in our method is the semantics-aware intent learning in Section 3.3) as the objective, the representations trained from it can be considered as the distilled information $\boldsymbol{u}^u$ that waits to adjust by $do(U = \boldsymbol{u}^u \odot C)$.

By far, we have the context information C for all aspects, where each $\boldsymbol{c_a} \in C$ represents context embedding of one aspect. We also have the refined intent representation $\boldsymbol{u}^u$ that waits to be adjusted from Eq. (13) as U. Next, we will detail the proposed causal intervention by providing implementation for Eq. (14).

The overall backdoor adjustment is achieved through:

$$\begin{aligned}
&P(Y \mid U, do(U = u)) - P(Y \mid do(U = 0)) \\
&= \sum_{C} \left(P\left(Y \mid do\left(U = \boldsymbol{u}^u \odot C\right)\right) - P\left(Y \mid do\left(U = 0\right)\right)\right) P\left(C = \boldsymbol{c_a}\right) \\
&= \frac{1}{N} \sum_{i=1}^{N} \left(P\left(\hat{\boldsymbol{y}}_{ui} \mid \boldsymbol{u}^u \odot C\right) - P\left(\hat{\boldsymbol{y}}_{ui} \mid \boldsymbol{u}^u\right)\right)
\end{aligned} \tag{15}$$

where each component in Eq. (15) is designed by:

- $C = \boldsymbol{c_a}$ indicates the confounder C is set as one context embedding $\boldsymbol{c_a}$.
- $P(Y|U, do(U = u)) = P\left(\hat{\boldsymbol{y}}_{ui} \mid \boldsymbol{u}^u \odot C\right)$. The selected context embedding C is concatenated with $\boldsymbol{u}^u$ by the element-wise product $\odot$, which serves as the adjustment value for the prediction of $\hat{\boldsymbol{y}}_{ui}$.
- $P(C = \boldsymbol{c_a})$ is the prior distribution of different item aspect, by defining $P(C = \boldsymbol{c_a}) = 1/N$, we can assume a uniform prior for the adjusted features, where N is the total number of aspect type.

We then utilize an inference strategy to adaptively fuse the prediction scores from the $P\left(\hat{\boldsymbol{y}}_{ui} \mid \boldsymbol{u}^u \odot C\right)$ and $P\left(\hat{\boldsymbol{y}}_{ui} \mid \boldsymbol{u}^u\right)$. Specifically, we first train the recommender model by $P\left(Y \mid do\left(\hat{\boldsymbol{y}}_{ui} = \boldsymbol{u}^u \odot C\right)\right)$ and $P\left(Y \mid do\left(\hat{\boldsymbol{y}}_{ui} = 0\right)\right)$ and obtain $\hat{\boldsymbol{y}}_C$ and $\hat{\boldsymbol{y}}$, respectively. Then, the adjusted prediction scores $\hat{\boldsymbol{y}}_C$ and unadjusted prediction score $\hat{\boldsymbol{y}}$ are automatically fused to regulate the impact of backdoor adjustment. We define a indicator function $\mathbf{I}_a$ that determines whether to include $\boldsymbol{c_a}$ into the user intent $\boldsymbol{u}^u$ or not.

$$\mathbf{I}_a := \begin{cases} \boldsymbol{c_a} & \text{if } \tanh\left(\hat{\boldsymbol{y}}_C - \hat{\boldsymbol{y}}\right) > 0 \\ \mathbf{1} & \text{if } \tanh\left(\hat{\boldsymbol{y}}_C - \hat{\boldsymbol{y}}\right) < 0 \end{cases} \tag{16}$$

where $\tanh(\hat{\boldsymbol{y}}_C - \hat{\boldsymbol{y}}) > 0$ indicates that the backdoor adjustment leads a positive impact on recommendation result by considering the aspect $\boldsymbol{c_a}$. Otherwise, $\boldsymbol{c_a}$ leads a negative impact, which should be removed from user intent representation. Based on Eq. (16), the semantic-aware user intent representation can be refined as follows:

$$\boldsymbol{e_a} = \boldsymbol{u}^u \odot \mathbf{I}_a \tag{17}$$

To define the unbiased loss function for observation $\hat{\boldsymbol{y}}_{ui}$, we aim to maximize the discrepancy between the final adjusted and the unadjusted representation $\boldsymbol{u}^u$ under the guidance of $\boldsymbol{e_a}$, that is,

$$\mathcal{L}_d = \underset{\bar{\theta}}{\arg\min} \sum_{(u,i,\boldsymbol{y}_{ui}) \in O} (\boldsymbol{y}_{ui}, f(\boldsymbol{u}, \boldsymbol{i}, \prod_{a}^{N} \boldsymbol{e_a})) \tag{18}$$

where $\boldsymbol{u}$, $\boldsymbol{i}$ are the ID embedding for user u and item i, $f(\cdot)$ is defined in Eq. (12), and $\boldsymbol{y}_{ui}$ is the ground-truth for user u and item i.

3.5 Optimization

Our model ultimately has three loss functions, i.e., $\mathcal{L}_d$ of unbiased loss function for preference score estimation given in Eq. (18), $\mathcal{L}_\theta$ of *Heterogeneous Skip-Gram* model given in Eq. (2), and the BPR loss for preference score prediction given in Eq. (13). To this end, the objective function of our CaDSI method could be derived as:

$$\mathcal{L} = \lambda_d \mathcal{L}_d + \lambda_\theta \mathcal{L}_\theta + \lambda_z \mathcal{L}_{BPR} + \mathcal{R}(\Omega) \tag{19}$$

where Ω represents the trainable parameters and $\mathcal{R}(\cdot)$ is a squared L_2 norm regularization term on Ω to alleviate the overfitting problem. λ_d, λ_θ, λ_z are trade-off hyper-parameters of the three separate loss functions respectively. During the training, we optimize the objective function in Eq. (19) via Adam algorithm [26].

4 EXPERIMENTS

To more thoroughly evaluate the proposed methd, experiments are conducted to answer the following research questions:

- (**RQ1**) How confoundeing bias caused by the context information is manifested in real-world recommendation datasets?
- (**RQ2**) How does our model perform compared with state-of-the-art models for top-N recommendation?
- (**RQ3**) How does key components in our model impact the recommendation performance? (i.e., disentanglement learning task, causal intervention)? How do hyper-parameters in our model impact recommendation performance?
- (**RQ4**) How does our model interprets user intents for recommendation?

We first present the experimental settings for good reproducibility, followed by answering the above four research questions.

4.1 Experimental Settings

4.1.1 Datasets

We evaluate our model on three public accessible datasets for top-N recommendation. The statistics of the datasets are summarized in Table 2 and the selected meta paths for all data sets are shown in Table 3. To ensure the quality of all the datasets, we use the core settings, i.e., we transform explicit ratings into implicit data, where each interaction between the user and item is marked as 0 or 1 indicating whether the user has rated the item or not; we retaining users and items with at least five interactions and each user has at least five friends for both of the datasets. In the training phase, each observed user-item interaction is treated as a positive instance, while we use negative sampling to randomly sample an unobserved item and pair it with the user as a negative instance.

TABLE 2
Statistics of three Datasets. Density of dataset is $\#Interactions/(\#Users \cdot \#Items)$, Avg.Degree of A is $\#Relation/\#A$, Avg.Degree of B is $\#Relation/\#B$.

Dataset (Density)	Node	Relation A-B	Avg.Degree of A/B
MovieLens-HetRec (4.0%)	#User(U): 2,113	#U-M: 855,598	#U/M: 405.0/84.6
	#Movie(M): 10,109	#U-U: 0	#U/U: 0/0
	#Actor(A): 38,044	#M-A: 95,777	#M/A: 9.5/2.5
	#Director(D): 4,031	#M-D: 10,068	#M/D: 1.0/2.5
	#Country(C): 72	#M-C: 10,109	#M/C: 1.0/140.0
	#Genre(G): 20	#M-G: 20,670	#M/G: 2.0/1033.5
Douban Book (0.27%)	#User(U): 13,024	#U-Bo: 792,062	#U/Bo: 60.8/35.4
	#Book(Bo): 22,347	#U-U: 169,150	#U/U: 13.0/13.0
	#Group(Gr): 2,936	#U-Gr: 1,189,271	#U/Gr: 91.3/405.1
	#Author(Au): 10,805	#Bo-Au: 21,907	#Bo/Au: 1.0/2.0
	#Publisher(P): 1,815	#Bo-P: 21,773	#Bo/P: 1.0/12.0
	#Year(Y): 64	#Bo-Y: 21,192	#Bo/Y: 1.0/331.1
Douban Movie (0.63%)	#User(U): 13,367	#U-M: 1,068,278	#U-M: 79.9/84.3
	#Movie(M): 12,677	#U-U: 4,085	#U/U: 1.7/1.8
	#Group(Gr): 2,753	#U-Gr: 570,047	#U/Gr: 42.7/207.1
	#Actor(A): 6,311	#M-A: 33,587	#M-A: 2.9/5.3
	#Director(D): 2,449	#M-D: 11,276	#M/D: 1.1/4.6
	#Type(T): 38	#M-T: 27,668	#M/T: 2.2/728.1

TABLE 3
The selected meta paths for three datasets in our work.

Dataset	Meta path Schemes
MovieLens-HetRec	UMU, UMAMU, UMDMU, UMCMU, UMGMU MUM, MAM, MDM, MCM, MGM
Douban Book	UBoU, UBoAuBoU, UBoPBoU, UBoYBoU, UBoAuBoU BoUBo, BoPBo, BoYBo, BoAuBo
Douban Movie	UMU, UMDMU, UMAMU, UMTMU MUM, MAM, MDM, MTM

4.1.2 Baselines

To demonstrate the effectiveness, we compare our model with four classes of methods: (I) conventional entangled CF methods; (II) graph-based entangled recommendation methods; (III) HIN enhanced entangled recommendation methods, which model user-item interaction with rich context information as HIN; (IV) disentangled recommendation methods, which disentangle user intents or item aspects with different mechanisms; (V) causal-based recommendation methods.

- **NeuMF** [27] (I): This method combines deep neural networks with Matrix Factorization (MF) method for modeling the user-item interactions.
- **GC-MC** [20] (II): The method organizes user behaviors as a graph, and employs one Graph Convolution Network (GCN) encoder to generate representations based on first-order connectivity.
- **NGCF** [21] (II): This adopts three Graph Neural Network(GNN) layers to model at most third-order connectivity on the user-item interaction graph.
- **LightGCN** [13] (II): This is a state-of-the-art graph-based recommendation method that learns user/item embeddings by linearly propagating them with neighborhood aggregation in the GCN component.
- **IF-BPR** [28] (III): This method leverages meta path based social relations derived from a HIN, and proposes a social recommendation method that can capture the similarity of users for top-N recommendation.
- **MCRec** [29] (III): This method leverages meta path based context with co-attention mechanism for top-N recommendation in HIN.
- **NeuACF** [30] (IV): This method disentangles multiple aspects of users and items with a deep neural network for recommendation in HIN.
- **MacridVAE** [8] (IV): This method disentangle user intents behind user behaviors, assuming that the coexistence of macro and micro latent factors affects user behaviors.
- **DGCF** [31] (IV): This is a state-of-the-art CF-based disentangled recommendation method, which disentangles latent factors of user intents by the neighbor routing and embedding propagation.
- **DICE** [11] (V): This is a state-of-the-art causal-based recommendation method, which aims at disentangling users' interest by controlling the conformity bias using causal embedding.

4.1.3 Evaluation Metrics

We adopt two popular metrics: Recall@K and Normalized Discounted Cumulative Gain(NDCG)@K to evaluate the top-K recommendation performance of our model. K is set as 20 by default. In the inference phase, we view the historical items of a user in the test set as the positive, and evaluate how well these items are ranked higher than all unobserved ones. The average results w.r.t. the metrics over all users are reported.

4.1.4 Parameter Settings

We implement all baseline models and our proposed CaDSI [1] model on a Linux server with Tesla P100 PCI-E 16GB GPU. For a fair comparison, datasets for implementing all models are split as train/test/validate set with a proportion of 80%/10%/10% of the dataset, while we optimize all models with Adam [26]. For a fair comparison, a grid search is conducted to choose the optimal parameter settings, e.g., dimension of user/item latent vector k_{MF} for matrix factorization-based models and dimension of embedding vector d for neural network-based models. The embedding size is initialized with the Xavier [32] and searched in $\{8, 16, 32, 64, 128, 256\}$. The batch size and learning rate are searched in $\{32, 64, 128, 512, 1024\}$ and $\{0.0005, 0.001, 0.005, 0.01, 0.05, 0.1\}$, respectively. The maximum epoch N_{epoch} is set as 2000, an early stopping strategy is performed. Moreover, we employ three hidden layers for the neural components of GC-MC NGCF, LightGCN, MCRec, NeuACF, MacridVAE and DGCF. The hyperparameter specifications of CaDSI are set as: latent intents number k as 4, iteration number of disentangling module $\boldsymbol{l}$ as 2, model depth of disentangling module L as 2, iteration number of causal intervention n as 140, and their influences are reported in Section 4.4.

4.2 Understanding Confounders (RQ1)

We initially conduct an experiment to understand to what extent the confounding bias exists in meta paths of real-world recommendation datasets. To this end, we aim to investigate the distribution of nodes among the same meta path. Intuitively, an unbiased HIN-based recommendation method should expect that, for a specific attribute, each

[1] Our code is currently shared on Github: https://github.com/AlfredYuisGood/CaDSI-Implementation-Code

user/item should hold an equal number of this attribute (i.e, interactions between nodes and attributes are likely to be evenly distributed). Thus, we investigate the confounding bias by analyzing the statistics of node-attribute interactions of meta paths in Douban Book. We randomly sample $n = 100$ books from Douban Book and extract their *Author*, *Publisher*, and *Year* attributes. By counting the connections between books and their attributes whose type belongs to *Author*, *Publisher*, and *Year*, respectively, we have the statistical results shown in Figure 4. The connected graphs in the left part of Figure 4 depict whether the book i connected with the selected attribute. The figures in the right part of Figure 4 shows the distributions of connected book numbers by a certain attribute, where the y-axis denotes the total amount of the connected books.

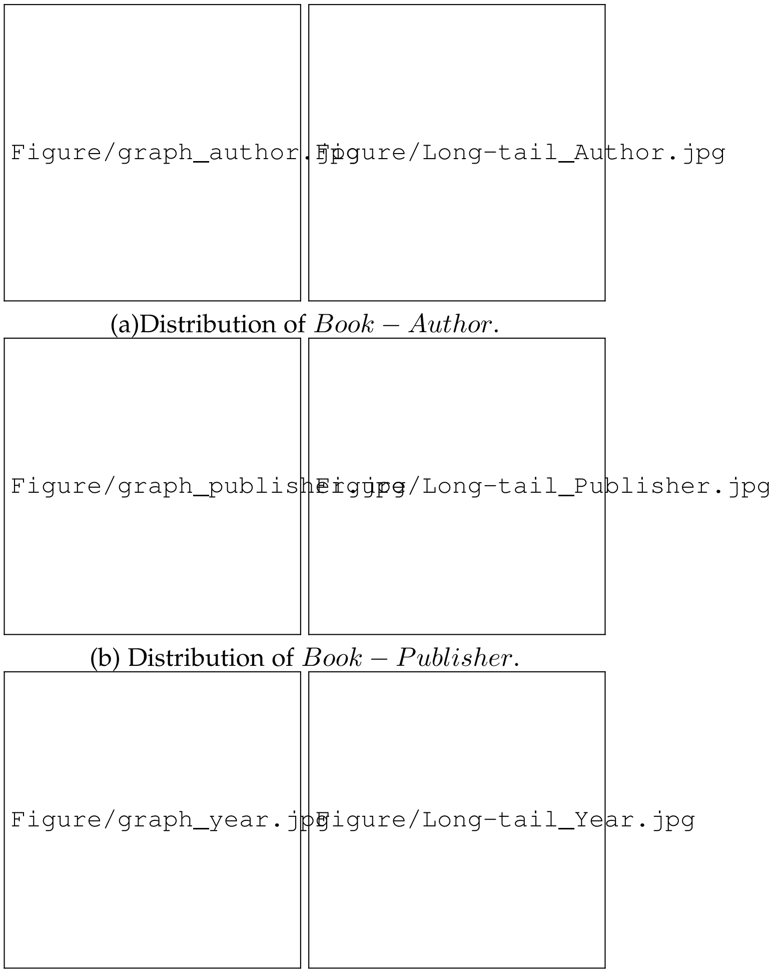
Apparently, attributes and books exhibit an unevenly distribution regarding their interactions: the attributes in dataset are partially observed, leaving a larger number of attributes to be unobserved. For example, for *Book-Author* meta path, there are a lot of books that do not connect with any node whose type is *Author*, which means lots of author attributes of books are missing. In conventional recommendation methods, the missing pattern of such attributes is ignored by either regarding them as outliers and padding them with random values, or treating the missing attributes as negative feedback. Such measure would preserve the confoundings bought by node attributes, degrading the recommendation ultimately.

Another finding is that, the distribution for book-attribute connection numbers is significantly skewed, it displays a long-tail phenomenon: the green vertical line separates the top 50% of connection numbers by popularity - these connections outweigh another 50% long tail connections to the right. For instance, in Figure 4 (a), authors in the *Book-Author* relation cumulatively connect with 90% more books than the long tail authors to the right. For *Book-Publisher* meta path, ideally, *Book-Publisher* has the one-to-one relation from book to the publisher, while every publisher has published an equal number of books. However, some publishers have published at most 510 books, while more than 90% of publishers only published fewer than 10 books. Such long-tail distribution can bias the users' interest on item aspects. i.e., recommendation methods tend to recommend those items that have the most frequent attribute, while users can only be exposed to those that are recommended.

4.3 Performance Comparison (RQ2)

We compare the top-K recommendation performance of CaDSI with ten recommendation baselines on three datasets: MovieLens-HetRec, Douban Book and Douban Movie. Table 4 demonstrates the performance comparison and we have the following observations:

- Our CaDSI consistently yields the best performance among all methods on three datasets. In particular, CaDSI improves over the strongest baselines w.r.t. Recall@20 by 23.5%, 22.7%, 13.6% , NDCG@20 by 11.9%, 18.8%, 3.8%, Recall@40 by 3.4%, 78.8%, 13.2% and NDCG@40 by 3.8%, 49.4%, 0.8% on MovieLens-HetRec, Douban Book and Douban Movie respectively. CaDSI outperforms all baseline methods on top-K recommendation task, which validates that the semantics-aware user intents representation can enhance the recommendation performance.
- In virtue of user-item interaction graph and meta paths, GNN-based (GC-MC, NGCF and LightGCN) and HIN-based (IF-BPR, MCRec) recommendation methods can achieve better performance than conventional MF methods (NeuMF) in most cases. However, they ignore controlling the bias existing in the context information. On the contrary, our CaDSI adoptes a principled causal inference way to easing such confounding bias. So it outperforms GNN-based and HIN-based recommendation method on both of the datasets. For instance, our CaDSI outperforms the most competitive HIN-based recommender IF-BPR w.r.t. Recall@20/Recall@40 by 24.2%/5.2% and NDCG@20/NDCG@40 by 29.2%/6.8% on MovieLens-HetRec.
- By performing unbiased disentanglement via semantics context, our CaDSI can infer user's potential interests of items. However, those user interests could not be well inferred from other disentangled rec-


Figure/graph_author.jpg
Figure/Long-tail_Author.jpg
Figure/graph_publisher.jpg
Figure/Long-tail_Publisher.jpg
Figure/graph_year.jpg
Figure/Long-tail_Year.jpg

(a)Distribution of $Book - Author$.

(b) Distribution of $Book - Publisher$.

(c) Distribution of $Book - Year$.

Fig. 4. The distributions of $Book - Author$, $Book - Publisher$ and $Book - Year$ of Douban Book dataset.

TABLE 4
Overall Performance Comparison: bold numbers are the improvement percentages; the best results are marked with *, strongest baselines are marked with underline.

	MovieLens-HetRec				Douban Book				Douban Movie			
	Recall@20	NDCG@20	Recall@40	NDCG@40	Recall@20	NDCG@20	Recall@40	NDCG@40	Recall@20	NDCG@20	Recall@40	NDCG@40
NeuMF	0.0434	0.0557	0.0665	0.0709	0.0339	0.0391	0.0641	0.0682	0.0460	0.0417	0.0708	0.0611
GC-MC	0.0336	0.0404	0.0653	0.0584	0.0458	0.0402	0.0675	0.0643	0.0448	0.0461	0.0602	0.0622
NGCF	0.0365	0.0508	0.0699	0.0615	0.0252	0.0301	0.0707	0.0691	0.0475	0.0498	0.0689	0.0642
LightGCN	0.0466	0.0155	0.0615	0.0498	0.0201	0.0225	0.0531	0.0568	0.0294	0.0331	0.0499	0.0578
IF-BPR	0.0546	0.0510	0.0727	0.0689	0.0396	0.0463	0.0628	0.0601	0.0483	0.0501	0.0652	0.0603
MCRec	0.0352	0.0195	0.0680	0.0677	0.0165	0.0294	0.0481	0.0507	0.0281	0.0336	0.0618	0.0629
NeuACF	0.0236	0.0308	0.0556	0.0684	0.0298	0.0201	0.0601	0.0579	0.0351	0.0438	0.0571	0.0623
MacridVAE	0.0454	0.0290	0.0661	0.0592	0.0309	0.0425	0.0691	0.0645	0.0489	0.0441	0.0729	0.0616
DGCF	0.0229	0.0589	0.0532	0.0708	0.0431	0.0502	0.0649	0.0663	0.0416	0.0527	0.0702	0.0628
DICE	0.0549	0.0499	0.0740	0.0703	0.0577	0.0608	0.0820	0.0799	0.0513	0.0389	0.0811	0.0636
Our model	0.0678*	0.0659*	0.0765*	0.0736*	0.0708*	0.0722*	0.1466*	0.1194*	0.0583*	0.0547*	0.0918*	0.0647*
%improv.	**23.5%**	**11.9%**	**3.4%**	**3.8%**	**22.7%**	**18.8%**	**78.8%**	**49.4%**	**13.6%**	**3.8%**	**13.2%**	**0.8%**

ommendation methods (NeuACF, MacridVAE and DGCF).

- Among the GNN-based (GC-MC, NGCF and LightGCN) and HIN-based recommenders (IF-BPR, MCRec) recommenders and disentangled recommenders (NeuACF, MacridVAE and DGCF), causal-based disentangled method (DICE) serves as the strongest baseline in most cases. This justifies the effectiveness of easing the counfounding bias in context information when estimating disentangled users' interests. However, DICE performs worse than our CaDSI, as it ignores rich semantics information in HIN, and fails to ingest semantics aspects when disentangling user interests.
- From movie recommendation datasets, we can find that the improvements on MovieLens-HetRec is bigger than that on Douban Movie. This is reasonable since Douban Movie is much more sparser than MovieLens-HetRec with sparsity rate 0.63% vs. 4.0%, respectively. However, our CaDSI has better performance than all baselines on Douban Movie, because it achieves unbiased evaluation on high-order connectivity and rich semantics. This indicates that CaDSI is robust to the very sparse dataset.

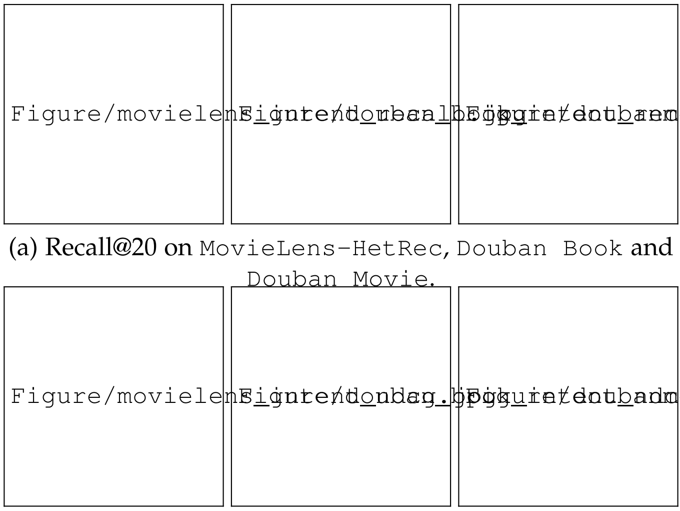
4.4 Study of CaDSI (RQ3)

Ablation studies on CaDSI are also conducted to investigate the rationality and effectiveness. Specifically, we first attempt to exploit how the *disentangled learning* and *causal intervention* affect our performance. Moreover, the stability of our approach's performance on top-K recommendation is validated as well.

We have one fixed parameter $n = 140$ (*cf.* Eq. (15)) which denotes the total number of causal intervention times. Three important hyperparameters k (*cf.* Eq. (4)), L (*cf.* Eq. (10)) and K (*cf.* Section 4.1.3) correspond to: the number of latent factors of user intents, the number of graph disentangling layers and the number of items in top-K recommendation list, respectively. Based on the hyperparameter setup in Section 4.1.4, for all questions listed above, we vary the value of one parameter while keeping the others unchanged.

4.4.1 Effect of Disentanglement Learning

The intent number k controls the total amount of user intents considered in our model, larger k stands for more fine-grained disentangled user intents. To study the influence, we vary k in the range of $\{1, 2, 4, 8, 16\}$ and show the corresponding performance comparison on MovieLens-HetRec Douban Book, Douban Movie in Figure 5. We have several observations.


(a) Recall@20 on MovieLens-HetRec, Douban Book and Douban Movie.

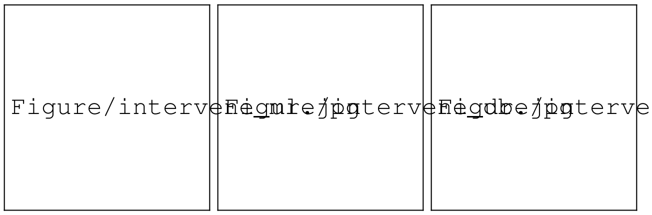
(b) NDCG@20 on MovieLens-HetRec, Douban Book and Douban Movie.

Fig. 5. The recommendation performance comparison under different latent user intent factors.

- Increasing the intent number from 1 to 16 can significantly enhances the performance, while CaDSI performs the worst when $k = 1$. This indicates learning the disentanglement of user intents is effective to capture the real user preferences towards items instead of coupling all preference together.
- The variations diverse across different datasets. For MovieLens-HetRec and Douban Movie, the performance of CaDSI increase steadily as the K value increases from 1 to 16, while the performance drops when k is set from 2 to 4 on Douban Book. One possible reason is that CaDSI should balance between too fine-grained disentangled intents and the adjustment from causal intervention, such balancing learning is more obvious when dataset size is lager.

4.4.2 Effect of Causal Intervention

To investigate whether CaDSI can get benefit from causal intervention, we study the performance of CaDSI by varying the iterations of causal intervention. Figure 6 summarizes the experimental results w.r.t. `MovieLens-HetRec` `Douban Book`, `Douban Movie` and we have the following observations:



(a) `MovieLens-HetRec`. (b) `Douban Book`. (c) `Douban Movie`.

Fig. 6. Impact of causal intervention on the recommendation performance of our CaDSI along with iterations.

- Clearly, the causal intervention mechanism renders our CaDSI a better recommendation performance: more iterations of causal intervention lead to the better recommendation performance before saturation on all datasets, e.g., the Recall@20 and NDCG@20 values generally increase along with training iterations in Figure 6.
- When training iterations reach to 130, 70 and 110 for `MovieLens-HetRec`, `Douban Book` and `Douban Movie`, respectively, the performance becomes relatively stable. Moreover, `Douban Book` requires less iteration times than the other two datasets. Intuitively, purchasing books is a much more simple behavior than choosing movies. Thus the user intents on book aspects are less diverse, leading to a quick convergence to the optimal interventional representations.
- Some fluctuations appear in the iteration process, especially on `MovieLens-HetRec` dataset. The size of `MovieLens-HetRec` is much smaller than the other two datasets, thus leading to an instability to intervention process due to the data sparsity. However, when carrying more iterations, small-size datasets such as `MovieLens-HetRec` can also yield satisfying results.

4.4.3 Effect of Multi-order Connectivity

Since CaDSI is benefited from the higher-order connectivity between complex interactions and context information, we investigate how connectivity degrees affect CaDSI. Specifically, we search the graph disentangling layer number L in the range of $\{1, 2, 3\}$, which correspond to first-order connectivity, second-order connectivity and third-order connectivity, respectively. We show the performance comparison in Table 5 and below are our observations.

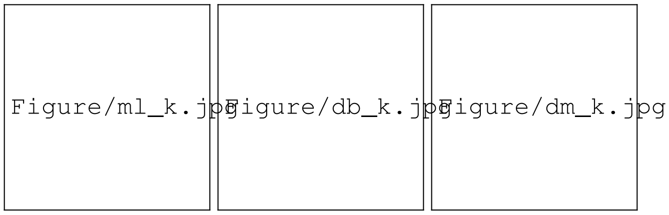
TABLE 5
Impact of multi-order connectivity (i.e., graph propagation layer number L) on `MovieLens-HetRec`, `Douban Book` and `Douban Movie`.

	MovieLens-HetRec		Douban Book		Douban Movie	
Layer number	Recall	NDCG	Recall	NDCG	Recall	NDCG
1	0.0505	0.0521	0.0651	0.0684	0.0583	0.0546
2	0.0672	0.0683	0.0712	0.0736	0.0596	0.0570
3	0.0611	0.0624	0.0682	0.0701	0.0562	0.0573

- More graph disentangling layers will collect more information form multi-hop neighbors from a holistic user-item interaction graph. Clearly, the performance of our CaDSI with layer number $L = 2$ is better than that with $L = 1$, since the second-order connectivity can capture significant collaborative signals with respect to users and items.
- When stacking more than 2 layers, the influence of multi-hop neighbors is small and the recommendation performance is degraded. This is reasonable since the informative signals of user-item interactions might introduce additional noises to the representation learning. This again emphasizes the importance of controlling the bias bought by context information.

4.4.4 Top-K Recommendation Performance

Based on the evaluation on Recall@K and NDCG@K, Figure 7 shows that CaDSI achieves the stable performance on top-K recommendation when K (i.e., the length of ranking list) varies from 10 to 80. This indicates that our CaDSI performs stably on top-K recommendation task and can recommend more relevant items within top-K positions when the ranking list length increases.



(a) `MovieLens-HetRec`. (b) `Douban Book`. (c) `Douban Movie`.

Fig. 7. Performance of CaDSI in terms of Recall@K and NDCG@K under different K.

4.5 Case Study and Visualization(RQ4)

We conduct experiments to get deep insights into the disentangled representations w.r.t. the disentanglement of the semantics of user intents, the representability and interpretability of the learned disentangled embedding. The case studies towards the disentanglement of the semantics of user intents are shown in Appendix A, the visualization result of the learned disentangled embedding is shown in Appendix B.

5 Related Work

In this section, we will introduce previous works related to ours from the following three aspects, including HIN enhanced representation, disentangled representation and causal inference for recommendation.

5.1 HIN Enhanced Representation

As a newly emerging direction, heterogeneous information network [14] is proved to be effective in modeling complex objects and providing rich semantics information to recommender systems [14], [33]. Many HIN-based recommendation methods achieve the-state-of-the-art performance [14]. For example, HeteMF [34] utilizes meta path based similarities as regularization terms in the MF model. HeteRec [35] learns meta path based latent features based on different types of entity relationships and proposes an enhanced personalized recommendation framework. SemRec [36] proposes a weighted HIN and designs a meta path based CF model to flexibly integrate heterogeneous information for a personalized recommendation. The effectiveness of HIN has been proved by a vast amount of HIN-based recommendation methods [14], thus, in our work, we value HIN in providing rich semantics information of user and item types. Despite the effectiveness, neither the enhanced graph-based nor HIN-based representations can disentangle users' intents by just presuming a uniform entangled embeddings behind behaviors. This can result in the poor interpretability of the developed recommendation methods. Thus, disentangle representation learning, which aims to learn factorized representations that separate and uncover latent explanatory factors behind the data [37], has recently received much attention in recommendation systems.

5.2 Disentangled Representation

Previous study has demonstrated that disentangled representations are more robust, i.e., counfounding bias are less likely to be preserved by uncovering latent factors. Ma et al. [8] propose to differentiate latent factors of learned user/item embeddings into macro and micro ones, thus the developed recommendation methods are less likely to mistakenly preserve the confounding of the factors. Moreover, the disentangle representation can provide rich semantics of users' preference, involving items' aspect information [8], [38] to users' behavior type information [39]. Thus, several works are proposed using disentangle representation learning to improve recommendation, for instance, DisHAN [40] learns disentangled aspect-aware user/item representations based on different meta path types in a HIN, these aspect-aware embeddings are then used to guide the top-N recommendation. Unfortunately, these aspect-aware disentangled embeddings only captured users' general taste on item aspects, however, failed to combine the specified item aspects with the real user intents, Parallelly, several works are conducted on modeling disentangled representations of users' intents, such as MacridVAE [8], DICE [11] and DGCF [31], the drawback is also distinct that they failed to combine the learned user intents with real-world item aspects, i.e., the user intents are predefined manually, such as "passing the time", short of providing meaningful information in a recommendation method. To sum up, the current studies on disentangle representation learning-enhanced recommendation either target at learning items' aspect-level representation [30], [40] or users' intents-level representation [8], [11], [31]. However, aforementioned approaches ignore bias stemmed from semantics information. To our knowledge, our approach is the first attempt to achieve interpretable and unbiased recommendation with disentangled embeddings for user intent.

5.3 Causal Methods for Debiasing

To the best of our knowledge, existing causal methods for recommendations aim at mitigating the effects of different bias rather than improving interpretability as in our work. Most existing works claim that the observational rating data suffers from *selection bias* [9], [41], *exposure bias* [6], [7], [42], [43] or *popularity bias* [3], [4], [10]. Following this paradigm, dominant approaches adopt two main strategies such as propensity-score [5], [6], [42] or causal embedding [7], [41], [43], to disentangle user interests from different types of bias. For instance, the method in [6] uses propensity score to re-weight the observational click data, with the aim of imitating the scenario that item is randomly exposed and alleviating the *exposure bias*. The work in [7] learns a uniform unbiased embeddings from partially observed user-item interactions via their decounfonded model. More recently, the work in [41] resorts to balance learning with a Middle-point Distance Minimization (MPDM) strategy to learn causal embeddings that are free from *selection bias*. Facing user conformity issue in recommendation, [3] relates such issue with *popularity bias*, and proposes to alleviate the popularity bias by learning disentangled embeddings of user interest. A few state-of-the-art works [4], [9], [10] inspect cause-effect of the bias generation and design a specific causal graph attributing the *exposure bias* to a confounder. For example, Li et al. [9] prove the social network to be a confounder that affects the user's rating and the exposure policy of the item to the user.

6 Conclusion and Future Work

In this paper, we have researched the confounding bias issue stemming from different aspects, and propose an unbiased and robust *Causal Disentanglement Semantics-Aware Intent Learning* (CaDSI) for recommendation. Our CaDSI is capable of providing semantics to fine-grained representations for disentangling user intents, meanwhile easing the bias stemming from unevenly distributed item aspects. We evaluate our CaDSI on three real-world recommendation datasets, with extensive experiments and visualizations demonstrate the robustness and interpretability of our semantics-aware user intent representation. In future work, we will explore the effect of different auxiliary information on the recommendation system using the intervention analysis in causal inference.

References

[1] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[2] R. Salakhutdinov, A. Mnih, and G. Hinton, "Restricted boltzmann machines for collaborative filtering," in *Proceedings of the 24th international conference on Machine learning*, 2007, pp. 791–798.

[3] Y. Zheng, C. Gao, X. Li, X. He, Y. Li, and D. Jin, "Disentangling user interest and conformity for recommendation with causal embedding," in *Proceedings of the Web Conference 2021*, 2021, pp. 2980–2991.

[4] W. Wang, F. Feng, X. He, X. Wang, and T.-S. Chua, "Deconfounded recommendation for alleviating bias amplification," *arXiv preprint arXiv:2105.10648*, 2021.

[5] A. Gruson, P. Chandar, C. Charbuillet, J. McInerney, S. Hansen, D. Tardieu, and B. Carterette, "Offline evaluation to make decisions about playlistrecommendation algorithms," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019, pp. 420–428.

[6] T. Schnabel, A. Swaminathan, A. Singh, N. Chandak, and T. Joachims, "Recommendations as treatments: Debiasing learning and evaluation," in *international conference on machine learning*. PMLR, 2016, pp. 1670–1679.

[7] Y. Wang, D. Liang, L. Charlin, and D. M. Blei, "The deconfounded recommender: A causal inference approach to recommendation," *arXiv preprint arXiv:1808.06581*, 2018.

[8] J. Ma, C. Zhou, P. Cui, H. Yang, and W. Zhu, "Learning disentangled representations for recommendation," *NeurIPS*, 2019.

[9] Q. Li, X. Wang, and G. Xu, "Be causal: De-biasing social network confounding in recommendation," *arXiv preprint arXiv:2105.07775*, 2021.

[10] Y. Zhang, F. Feng, X. He, T. Wei, C. Song, G. Ling, and Y. Zhang, "Causal intervention for leveraging popularity bias in recommendation," *arXiv preprint arXiv:2105.06067*, 2021.

[11] Y. Zheng, C. Gao, X. Li, X. He, Y. Li, and D. Jin, "Disentangling user interest and conformity for recommendation with causal embedding," in *Proceedings of the Web Conference 2021*, ser. WWW '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 2980–2991. [Online]. Available: https://doi.org/10.1145/3442381.3449788

[12] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 974–983.

[13] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "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*, 2020, pp. 639–648.

[14] C. Shi, Y. Li, J. Zhang, Y. Sun, and S. Y. Philip, "A survey of heterogeneous information network analysis," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 1, pp. 17–37, 2016.

[15] J. Pearl, *Causality*. Cambridge university press, 2009.

[16] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 2017, pp. 135–144.

[17] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[18] L. Bottou, "Stochastic gradient descent tricks," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 421–436.

[19] S. Weisberg, *Applied linear regression*. John Wiley & Sons, 2005, vol. 528.

[20] R. v. d. Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," *arXiv preprint arXiv:1706.02263*, 2017.

[21] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, 2019, pp. 165–174.

[22] S. Sharma and S. Sharma, "Activation functions in neural networks," *Towards Data Science*, vol. 6, no. 12, pp. 310–316, 2017.

[23] S. Rendle, "Factorization machines," in *2010 IEEE International Conference on Data Mining*, 2010, pp. 995–1000.

[24] S. Zhang, Z. Han, Y.-K. Lai, M. Zwicker, and H. Zhang, "Stylistic scene enhancement gan: mixed stylistic enhancement generation for 3d indoor scenes," *The Visual Computer*, vol. 35, no. 6, pp. 1157–1169, 2019.

[25] D. Lian, Q. Liu, and E. Chen, "Personalized ranking with importance sampling," in *Proceedings of The Web Conference 2020*, 2020, pp. 1093–1103.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[27] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 173–182.

[28] J. Yu, M. Gao, J. Li, H. Yin, and H. Liu, "Adaptive implicit friends identification over heterogeneous network for social recommendation," in *Proceedings of the 27th ACM international conference on information and knowledge management*, 2018, pp. 357–366.

[29] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging metapath 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*, 2018, pp. 1531–1540.

[30] X. Han, C. Shi, S. Wang, S. Y. Philip, and L. Song, "Aspect-level deep collaborative filtering via heterogeneous information networks." in *IJCAI*, 2018, pp. 3393–3399.

[31] X. Wang, H. Jin, A. Zhang, X. He, T. Xu, and T.-S. Chua, "Disentangled graph collaborative filtering," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 1001–1010.

[32] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[33] Y. Sun and J. Han, "Mining heterogeneous information networks: a structural analysis approach," *Acm Sigkdd Explorations Newsletter*, vol. 14, no. 2, pp. 20–28, 2013.

[34] X. Yu, X. Ren, Q. Gu, Y. Sun, and J. Han, "Collaborative filtering with entity similarity regularization in heterogeneous information networks," *IJCAI HINA*, vol. 27, 2013.

[35] X. Yu, X. Ren, Y. Sun, Q. Gu, B. Sturt, U. Khandelwal, B. Norick, and J. Han, "Personalized entity recommendation: A heterogeneous information network approach," in *Proceedings of the 7th ACM international conference on Web search and data mining*, 2014, pp. 283–292.

[36] C. Shi, Z. Zhang, P. Luo, P. S. Yu, Y. Yue, and B. Wu, "Semantic path based personalized recommendation on weighted heterogeneous information networks," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 2015, pp. 453–462.

[37] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[38] Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu, and S. Ma, "Explicit factor models for explainable recommendation based on phrase-level sentiment analysis," in *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, 2014, pp. 83–92.

[39] B. Liu, A. Sheth, U. Weinsberg, J. Chandrashekar, and R. Govindan, "Adreveal: Improving transparency into online targeted advertising," in *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, 2013, pp. 1–7.

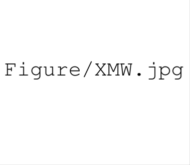
[40] Y. Wang, S. Tang, Y. Lei, W. Song, S. Wang, and M. Zhang, "Disenhan: Disentangled heterogeneous graph attention network for recommendation," in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 2020, pp. 1605–1614.

[41] L. Yao, S. Li, Y. Li, M. Huai, J. Gao, and A. Zhang, "Representation learning for treatment effect estimation from observational data," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

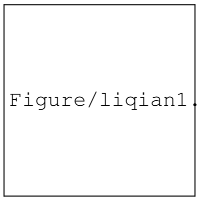
[42] D. Liang, L. Charlin, and D. M. Blei, "Causal inference for recommendation," in *Causation: Foundation to Application, Workshop at UAI*. AUAI, 2016.

[43] S. Bonner and F. Vasile, "Causal embeddings for recommendation," in *Proceedings of the 12th ACM Conference on Recommender Systems*, 2018, pp. 104–112.

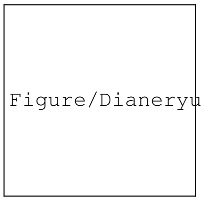
[44] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.



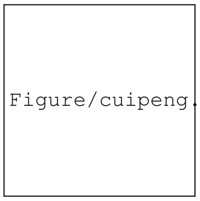
Xiangmeng Wang has been a Ph.D. student at the School of Computer Science, Faculty of Engineering and Information Technology, University of Technology Sydney (UTS). She received her MSc degree in Computer Application Technology from Shanghai University. Her general research interests lie primarily in explainable artificial intelligence, data analysis, and causal machine learning. Her papers have been published in the top-tier conferences and journals in the field of machine learning.



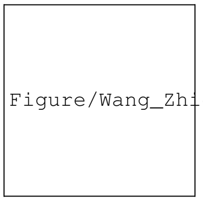
Qian Li is a Lecturer at School of Electrical Engineering, Computing and Mathematical Sciences, Curtin University, Perth, Australia. She has been a Postdoc Research Fellow at the Faculty of Engineering and Information Technology, University of Technology Sydney (UTS). She received her Ph.D. in Computer Science from the Chinese Academy of Science. Her general research interests lie primarily in optimization algorithms, topological data analysis, and causal machine learning. Her papers have been published in the top-tier conferences and journals in the field of machine learning and computer vision.



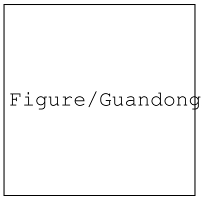
Dianer Yu has been a Postgraduate IT student at the School of Computer Science, Faculty of Engineering and Information Technology, University of Technology Sydney (UTS). He received his BSc degree of IT from University of Technology Sydney. His general research interests lie primarily in data mining, causal model for recommendation and explainable machine learning. He has been awarded as Postgraduate Dean's List during the Postgraduate period.



Peng Cui is an Associate Professor at Tsinghua University. He received his Ph.D. degree in computer science in 2010 from Tsinghua University. He has vast research interests in data mining, multimedia processing, and social network analysis. Until now, he has published more than 20 papers in conferences such as SIGIR, AAAI, ICDM, etc. and journals such as IEEE TMM, IEEE TIP, DMKD, etc. Now his research is sponsored by National Science Foundation of China, Samsung, Tencent, etc. He also serves as Guest Editor, Co-Chair, PC member, and Reviewer of several high-level international conferences, workshops, and journals.



Zhichao Wang received his Ph.D. degree from Department of Automation, Tsinghua University. He was a Research Fellow at University of New South Wales. His research interests lie in the optimization for machine learning and stochastic modelling.



Guandong Xu is a Professor in the School of Computer Science and Advanced Analytics Institute at University of Technology Sydney. He received MSc and BSc degree in Computer Science and Engineering, and PhD in Computer Science. He currently heads the Data Science and Machine Intelligence Lab, which consists of 15+ members of academics, research fellows and HDR students. From Nov 2019, he directs the newly established Smart Future Research Centre, which is an across-disciplines industry engagement and innovation platform for AI and Data Science Application towards smart wealth management and investment, energy, food, water, living, and city.

Appendix A
Case Studies

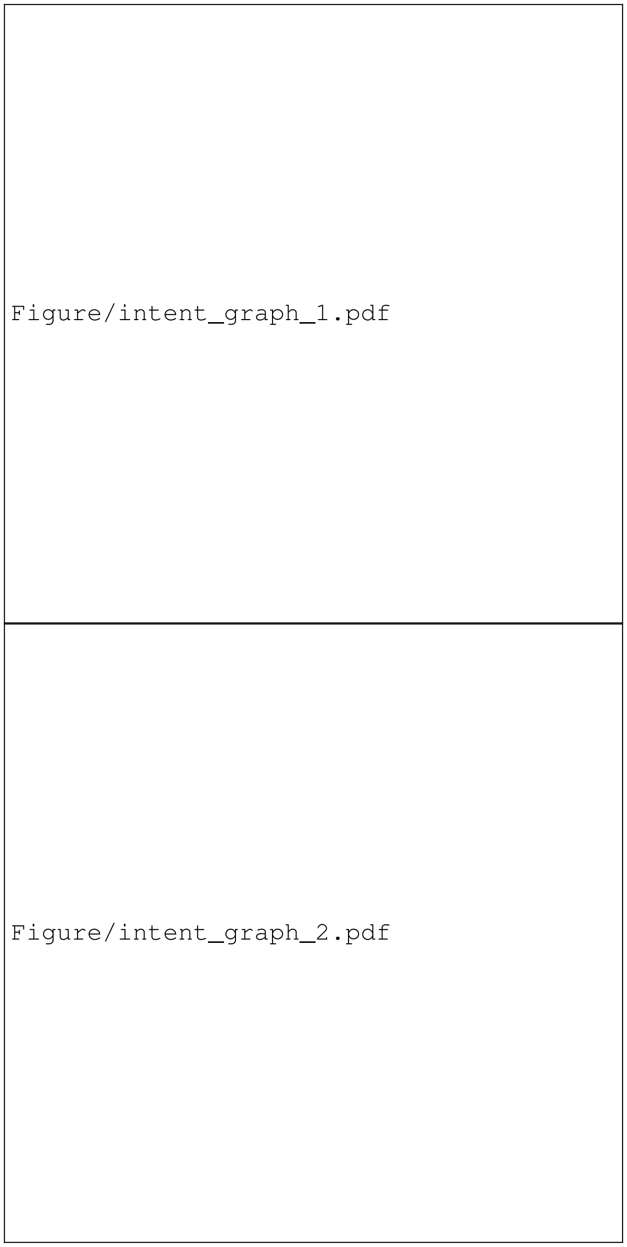


Fig. 8. Visualization of the disentangled user intent graphs based on score matrices. user-item interactions with highest scores are marked in solidlines; item attributes with the same values are highlighted in red.

We first conduct an experiment to understand the disentanglement of user intents by our CaDSI, then explore whether such intent related to real-world item semantics. We select an user $u2972$ from `Douban Book` and learn its interaction scores $\mathbf{S}(u,i)$ (*cf.* Eq. (9)) with his/her historical interacted items under our CaDSI. The user intents factor $k = 4$ indicates four distinct user intents. Thereafter, we randomly select four items from the interaction score matrices. For each interaction under different k, we mark the interaction scores with the highest confidence with solid lines and couple the certain item attributes below them. Figure 8 shows the visualization results and we have the following findings:

- Jointly analyzing intent-aware user-item interaction graphs, we can see user preference differs across each graphs, reflected by different interaction scores in each intent-aware graph. For example, $u2972$ interacts with $i12047$ with a preference score of 1.43 under intent k_1, while the score changes to 1.79 under intent k_2. This demonstrates the importance of disentangling user intents in recommendation scenario.
- We thereafter couple item attributes to investigate whether user intents are related to item semantics. It can be seen that different fine-grained user intents are highly consistent with high-level item semantics. For instance, intent k_3 contributes mostly to interactions $(u2972, i14892)$ and $(u2972, i18)$, which suggests its high confidence as being the intents behind these behaviors. When switch to item attributes of $i12047$ and $i18$, one highlighted item attribute *Author* with the same id can be found, reflecting the reason why $u2972$ chose to interact with $i12047$ and $i18$. This demonstrated that our CaDSI, which aims at disentangling user intents meanwhile assign specific item semantics to the learned intents, is effective in the disentanglement of user intents towards item aspects.

APPENDIX B
VISUALIZATION

Figure/case_2_abstract.jpg

Fig. 9. 2-dimensional t-SNE projections of the 128-dimensional embeddings of 100 users from `Douban Book` dataset.

We randomly select 100 users from `Douban Book` dataset, and implement CaDSI on the dataset to output the 128-dimensional semantics-aware user intent embedding $\boldsymbol{e}$. For visualization purposes, we use `t-SNE` [44] to map high-dimensional user intent representation $\boldsymbol{e}$ to 2-dimensional vectors. Following the parameter settings in Section 4.1.4, we set the user intent factors $k = 4$. Figure 9 shows the visualization result.

We notice that the projections are capable of distinguishing four discernible clusters of users, and the cluster number is consistent with our pre-defined latent user intent factors k. This indicates that CaDSI is able to group users of the same intent closely based on the distances among users' embeddings. Meanwhile, each cluster is well-separated from others, further demonstrating the robust representation of CaDSI.

Furthermore, we extract two users termed 1377903 and $G.Frankenstein$ and show their historical interaction abstracts in the right of Figure 9. Analyzing these abstracts, we can see that our CaDSI is also capable of grouping each user whose interests are on the same item attributes. Such as 1377903 and $G.Frankenstein$, both users arrange items with similar attributes close to each other and dissimilar ones distant from each other. In summary, the visualizations intuitively demonstrate CaDSI's novel capability to discover, model, and capture the underlying semantics and structural relationships between multiple item aspects and user intents in heterogeneous networks.