

Multi-View Intent Disentangle Graph Networks for Bundle Recommendation

Sen Zhao^{1,3}, Wei Wei^{*1,3}, Ding Zou^{1,3}, Xianling Mao²

¹ Cognitive Computing and Intelligent Information Processing (CCIIP) Laboratory, School of Computer Science and Technology, Huazhong University of Science and Technology

² Beijing Institute of Technology

³ Joint Laboratory of HUST and Pingan Property & Casualty Research (HPL)
{senzhaow, weiw, m202173662}@hust.edu.cn, maoxl@bit.edu.cn

Abstract

Bundle recommendation aims to recommend the user a bundle of items as a whole. Previous models capture the user's preferences on both items and the association of items. Nevertheless, they usually neglect the diversity of the user's intents on adopting items and fail to disentangle the user's intents in representations. In the real scenario of bundle recommendation, a user's intent may be naturally distributed in the different bundles of that user (Global view), while a bundle may contain multiple intents of a user (Local view). Each view has its advantages for intent disentangling: 1) From the global view, more items are involved to present each intent, which can demonstrate the user's preference under each intent more clearly. 2) From the local view, it can reveal the association among items under each intent since items within the same bundle are highly correlated to each other. To this end, we propose a novel model named Multi-view Intent Disentangle Graph Networks (MIDGN), which is capable of precisely and comprehensively capturing the diversity of the user's intent and items' associations at the finer granularity. Specifically, MIDGN disentangles the user's intents from two different perspectives, respectively: 1) In the global level, MIDGN disentangles the user's intent coupled with inter-bundle items; 2) In the Local level, MIDGN disentangles the user's intent coupled with items within each bundle. Meanwhile, we compare the user's intents disentangled from different views under the contrast learning framework to improve the learned intents. Extensive experiments conducted on two benchmark datasets demonstrate that MIDGN outperforms the state-of-the-art methods by over 10.7% and 26.8%, respectively. The implementation of our proposed models is publicly available at <https://github.com/Snnzhao/MIDGN.git>.

Introduction

Recently, the blossom of Internet Technology has revolutionized our routine activities by providing a tremendous amount of valuable contents, which however lead to the serious information overload problem (Zhao et al. 2019; Wei et al. 2019). To alleviate this problem, recommendation system (Zheng et al. 2010) has been an effective tool to help the user quickly discover items of potential interest to themselves. In general, conventional recommendation systems

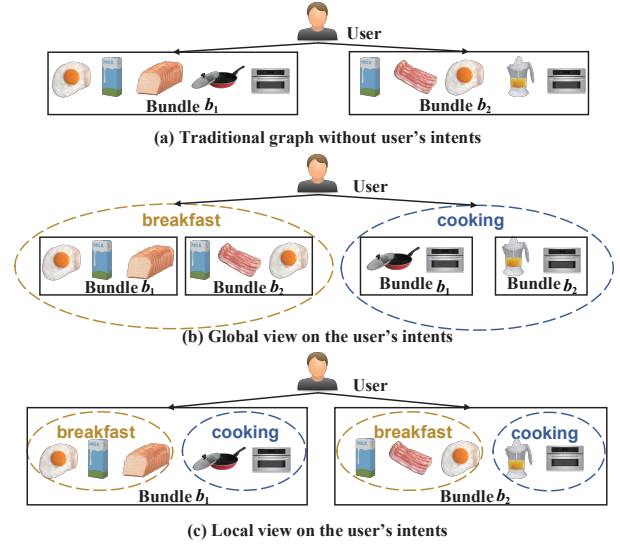


Figure 1: An example of the user's multi-intent pattern for bundle recommendation.

(Wu et al. 2019; Wang et al. 2020b,c,e,d) aim to recommend individual items to users, but in practice (e.g., E-commerce), items bundling is also a widely-adopted solution to increase the exposure of items that are seldom purchased in isolation. To this end, the problem of bundle recommendation (BR) (Zhu et al. 2014) is proposed to recommend a bundle of items that the user has interest to purchase together.

Similar to conventional recommendation system, learning the user's preferences is the main theme for bundle recommendation. Existing methods mainly focus on capturing the user's preferences on both items and the bundle of items. Liu et al. (2014) and Cao et al. (2017) simultaneously utilize the user's interactions with both items and bundles under the Bayesian Personalized Ranking (BPR) (Rendle et al. 2014) framework. Chen et al. (2019) jointly model user-bundle and user-item interactions in a multi-task manner, but they may neglect the user's preference over the association of items within bundles, which is also of crucial importance to bundle recommendation. Chang et al. (2020) unify user-bundle-item affiliation into a heterogeneous graph and uses graph convolution networks (GCN) to learn the user

* Corresponding author.

and bundle’s representation, capturing item level semantics, however these works may neglect the diversity of the user’s intents on adopting items and fail to disentangle the user’s intents in representations. Wang *et al.* (2020a) consider user-item relationships at the finer granularity of the user’s intents and generate disentangled representations for the item recommendation problem. Nevertheless, the real scenarios of bundle recommendation are more complicated (as shown in Figure 1), a user’s intent may be naturally distributed in the different bundles of that user. (*Global view*), while a bundle may contain multiple intents of a user (*Local view*). There are two obvious drawbacks when neglecting the diversity of the user’s intents: 1) from the global view, the inter-bundle items under different intents can interfere with each other. As illustrated in Figure 1(b), the egg can be noisy for constructing the intent of cooking. 2) from the local view, the association of inter-intent items may have a negative rating on the user’s interest for each bundle. For the example in Figure 1(b), bundle b_1 may not be recommended because bread and pan are rarely purchased together. Each view has its own advantages for intent disentangling: 1) In the global view, more items are involved to present each intent, which can demonstrate the user’s preference under each intent more clearly. 2) The local view can reveal the association among items under the user’s each intent since the items within the same bundle are highly correlated to each other. Disentangling the user’s intent from both views, we can better represent the user’s preference and the association among items at a finer grain of the user’s intent.

In this work, we propose a novel model, Multi-view Intents Disentangle Graph Networks (MIDGN), to disentangle the user and bundle representations at the granularity of the user’s intents. Specifically, we disentangle the representations of the user and the bundle into chunks, each chunk represents a latent intent. A graph neural network equipped with neighbour routing is applied both to disentangle the user-item and the bundle-item graph, where such two types of graphs can provide different views of users’ intent for recommendation. Furthermore, a contrast learning module comparing the user’s intents under different views is also employed for enhancing the learning of user intent representations. Extensive experiments conducted on NetEase and YouShu demonstrate that our proposed model MIDGN outperforms the state-of-the-art methods by over 10%.

In a nutshell, this work makes the following contributions:

- We emphasize the importance of disentangling the user’s intent in bundle recommendation problem and explore two views for the user’s intents disentangling.
- We propose the model MIDGN, which disentangles the user and bundle’s representation by modeling and contrasting the user’s intents under global and local views.
- We conduct extensive experiments on two benchmark datasets and MIDGN achieves over 10 percents improvement over the state-of-art methods.

Related Work

The early work (Chen *et al.* 2005; Garfinkel *et al.* 2006) on bundle recommendation mainly focus on mining a group

of common items from historical user-item interactions for users to consume together, while ignoring the relevance of the selected items. For example, (Liu *et al.* 2011; Xie, Lakshmanan, and Wood 2010) formalize the bundle recommendation task as a (linear) knapsack problem and no item-to-item dependency is considered. (Parameswaran, Venetis, and Garcia-Molina 2011) models the curriculum recommendation as a *constraint satisfaction problem* and make use of cross-item dependencies as general hard-constraints for recommendation. However, previous bundle recommendation methods are incapable of discovering the items of common interest, and thus cannot accurately capture a use’s personal preference for recommendation.

In recent years, many studies have been devoted for bundle recommendation, which can be roughly categorized into two classes, (1) Intra-basket recommendation (Le, Lauw, and Fang 2017), which recommends items to be added to the current basket. There exist several attempts on this problem, for example, Le *et al.* (2017) propose a factorization based model (named CBFM) to identify the correlations of items within basket to integrate constraints for the given basket with similar intent. Liu *et al.* (2017) employ a probabilistic graphical model to infer users’ preferences over bundle via introducing two latent factors, *i.e.*, node-type (items) and edge-type (the associations between pairs of items). (2) Inter-basket (*a.k.a.*, next-basket), which offers multiple items for sale as a bundle (*a.k.a.*, bundle list recommendation (Sar Shalom *et al.* 2016; Bai *et al.* 2019)). Generally, bundle recommendation is formalized as a top-K optimization problem, which can be solved with learning-to-rank based approaches. For instance, Xie *et al.* (2014) take account of sampling and preference elicitation based method to generate top-K bundles for users, in which an additive utility function with a combination of the corresponding feature values is employed for tackling the problem, but may fail to capture the complicated dependencies of user-item-bundle interactions. Further efforts are conducted to capture the complicated dependencies. Cao *et al.* (2017) simultaneously utilize the users’ interactions with both items and bundles under the Bayesian Personalized Ranking (BPR) (Rendle *et al.* 2014) framework. Chen *et al.* (2019) jointly model user-bundle and user-item interactions in a multi-task manner. These methods, however, fail to learn multi-hop connectivity from user-bundle-item interaction and neglect user’s preference over the association of items within bundles which is also crucial for bundle recommendation.

Recently, graph convolution networks (GCN) (Kipf and Welling 2016) has been extensively used for recommendation due to its superior ability in learning of the graph structure. Berg *et al.* (2017) first apply GCN to recommendation to factorize several rating matrices. Wang *et al.* (2019) employ GCN to encode the collaborative signal in the form of high-order connectivities. Graph convolution networks are also adopted to solve the problem of bundle recommendation. Liu *et al.* (2020) designs three types of aggregators with GCN specifically for users, bundles and items. Chang *et al.* (2020) unify user-bundle-item affiliation into a heterogeneous graph and uses GCN to learn user and bundle representation capturing item level semantics. These works, how-

ever, neglect the diversity of user’s intents on adopting items and fail to disentangle user’s intents in representations.

Disentangling user’s intents has been a topic in the problem of item recommendation. Wang *et al.* (2020a) considers user-item relationships at the finer granularity of user’s intents and generates disentangled representations. But for the problem of bundle recommendation, where the user’s intents are distributed inter-and intra-bundles, disentangling user’s intents is more complicated and new methods should be studied.

Preliminaries

Let $\mathcal{U} = \{u_1, u_2, \dots, u_M\}$ be the user set, and $\mathcal{B} = \{b_1, b_2, \dots, b_O\}$ and $\mathcal{I} = \{i_1, i_2, \dots, i_N\}$ be the associated bundle set and item set, where M, O, N denote the number of users, bundles and items. According to the history of bundles user consumed, we can define the user-bundle interaction matrix, bundle-item affiliation matrix and user-item interaction matrix as $\mathbf{Y}_{M \times O} = \{y_{ub} | u \in \mathcal{U}, b \in \mathcal{B}\}$, $\mathbf{H}_{O \times N} = \{h_{bi} | b \in \mathcal{B}, i \in \mathcal{I}\}$ and $\mathbf{R}_{M \times N} = \{r_{ui} | u \in \mathcal{U}, i \in \mathcal{I}\}$, in which $y_{ub} = 1$, $h_{bi} = 1$ and $r_{ui} = 1$ means user u once interacted bundle b , bundle b contains item i , and user u once bought item i , respectively.

The task of bundle recommendation is to predict the probability of user u potentially interacting with a given bundle never seen before. Specifically, our goal is to learn a predict function $\hat{y} = \mathcal{F}(u, b | \theta, \mathbf{H})$, where \hat{y} is the estimation probability and θ implies the parameters of function \mathcal{F} .

The Proposed Model

In this section, we present Multi-view Intent Disentangle Graph Networks (MIDGN)(shown in Figure 2), which is composed of four different components: 1) *Graph disentangling module*, which disentangles user-item interactions and bundle-item interactions coupling with the user’s intents from *global* and *local* perspectives, respectively; 2) *Cosss-view propagating module* that propagates collaborative signal coupling with the user’s intents under different views through user-bundle interactions; 3) *Intent contrasting module*, which employs InfoNCE (Oord, Li, and Vinyals 2018) to encourage the correlation of the user’s intents under different views; and *predicting module*, which predicts the rating with the learned user and bundle embeddings.

Graph Disentangling Module

In this module, we first slice each user/bundle embedding into K chunks and couple each chunk with the user’s one intent. Then a graph neural networks incorporated with neighbor routing mechanism is devised to disentangle the user-item/bundle-item graph and refine the intent-aware user/bundle representation.

Initialization of intent-aware embeddings and graphs.

We assume the user has K intents and slice each user/bundle’s embedding into K chunks in different feature space. Coupling with each intent, a chunk of the user and bundle’s embedding is independently initialized respectively. Formally, the user and bundle’s embedding are represented

as:

$$\begin{aligned} \mathbf{u} &= (\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_K), \\ \mathbf{b} &= (\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_K), \end{aligned} \quad (1)$$

where $\mathbf{u} \in R^d$ and $\mathbf{b} \in R^d$ are embedding of the user and bundle. $\mathbf{u}_k \in R^{\frac{d}{K}}$, $\mathbf{b}_k \in R^{\frac{d}{K}}$ are the embedding chunks of the user and bundle under the k -th intent, respectively. As each item is bought by the user under one intent, there is no need to divide it’s embedding. So the item embedding is presented as $\mathbf{i} \in R^{\frac{d}{K}}$ and randomly initialized.

We assume each intent k is likely to be the reason why the user/bundle connects with a certain subset of its neighbour items. To present user/bundle-item interactions motivated by different intents, a set of intent-aware graphs $\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_K\}$ is built. For each intent-aware graph \mathcal{G}_k , a weighted adjacent matrix \mathbf{A}_k is built, where $\mathbf{A}_k(c, i)$ denotes the confidence that the interaction between c and item i is motivated by the k -th intent. Here c^1 is a uniform placeholder for user symbol u and bundle symbol b . Intuitively, each interaction within user/bundle-item graph is associated with the user’s K intents to different degree as follows:

$$\mathbf{A}(c, i) = (\mathbf{A}_1(c, i), \mathbf{A}_2(c, i), \dots, \mathbf{A}_K(c, i)), \quad (2)$$

We uniformly initialize the confidence of each interaction under the user’s intents as:

$$\mathbf{A}(c, i) = (1, 1, \dots, 1), \quad (3)$$

which denotes intents contributes equally to the each interaction.

Intent-aware interaction graph disentangling. Each intent k is specialized with embedding chunk \mathbf{c}_k as well as a specific interaction graph \mathcal{G}_k . When describe user/bundle c under intent k , we should construct \mathbf{c}_k with neighbours that connect to user/bundle c due to intent k . To realize this, a graph disentangling layer is employed to user-item and bundle-item graph as follows:

$$\mathbf{e}_{ck}^{(1)} = g(\mathbf{c}_k, \{\mathbf{i}, \mathbf{i} \in \mathcal{N}_c\}), \quad (4)$$

where $g(\cdot)$ denotes the graph disentangling layer. $i \in \mathcal{N}_c$ denotes items that have interaction with user/bundle c . $\mathbf{e}_{ck}^{(1)}$ is to capture the collaborative signal related to the user’s k_{th} intent within node c ’s neighbor. The super-index (1) denotes the first-order neighbors.

As illustrated in Figure 2, the neighbor routing mechanism is adopted in the graph disentangling module, which iteratively updates the user/bundle’s embedding chunks \mathbf{c}_k and the adjacent matrix \mathbf{A}_k for graph \mathcal{G}_k . For each iteration, \mathbf{c}_k^t and \mathbf{A}_k^t are used to memorize the update of \mathbf{c}_k and \mathbf{A}_k .

Update within each iteration. For each interaction (c, i) , we have its confidence under K intents $\{\mathbf{A}_k^t(c, i) | \forall k \in \{1, 2, \dots, K\}\}$. To calculate it’s distribution over intents, softmax function is applied to the confidence:

$$\tilde{\mathbf{A}}_k^t(c, i) = \frac{\exp \mathbf{A}_k^t(c, i)}{\sum_{k'=1}^K \exp \mathbf{A}_{k'}^t(c, i)}, \quad (5)$$

¹This paper uses symbol c as a uniform placeholder for user symbol u and bundle symbol b , since the calculation and formulation for the user and bundle in this article are similar in many scenarios

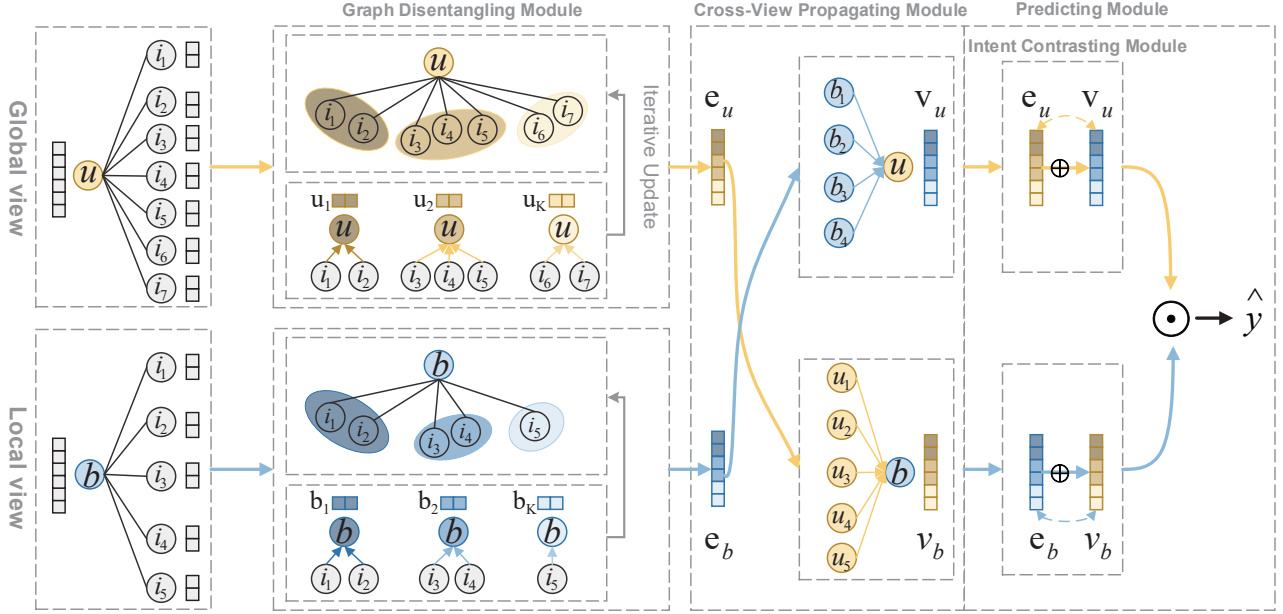


Figure 2: The framework of the proposed MIDGN. It contains four modules: (a) *Graph disentangling module*, which disentangles user-item and bundle-item interactions coupling with the user’s intents under global and local view respectively; (b) *Cross-view propagating module* that propagates collaborative signal coupling with the user’s intents under different views; (c) *Intent contrasting module* to encourage the correlation of the user’s intents under different views and *Predicting module*. (Best view in color)

which indicates the extent to which interaction (c, i) is motivated by each intent.

Then the embedding propagation is employed to each intent-aware graph \mathcal{G}_k , to update node c ’s embedding chunk under different intents with its neighbor items. Specifically, the weighted aggregation is defined as:

$$\mathbf{c}_k^t = \sum_{i \in \mathcal{N}_c} \frac{\tilde{\mathbf{A}}_k^t(c, i)}{\sqrt{D_t^k(c) \cdot D_t^k(i)}} \cdot \mathbf{i}, \quad (6)$$

where $D_t^k(c) = \sum_{i \in \mathcal{N}_c} \tilde{\mathbf{A}}_k^t(c, i)$ and $D_t^k(i) = \sum_{c \in \mathcal{N}_i} \tilde{\mathbf{A}}_k^t(c, i)$ are degrees of node c and item respectively. This normalization is to deal with the influence of varying numbers of nodes on the training process.

Then for each node c , we should update its confidence of connections with items $\mathbf{A}_k^t(c, i)$. To realize this, the confidence $\mathbf{A}_k^t(c, i)$ of interaction (c, i) under each intent is updated through:

$$\mathbf{A}_k^{t+1}(c, i) = \mathbf{A}_k^t(c, i) + \mathbf{c}_k^t \cdot \mathbf{i}. \quad (7)$$

If the item’s embedding \mathbf{i} are similar with the node c ’s embedding chunk \mathbf{c}_k^t under intent k , the confidence of their connection $\mathbf{A}_k(c, i)$ will be enhanced.

Layer combination. \mathbf{e}_{ck}^1 involve one-hop information within node c ’s neighbour. To incorporate multi-hop, more disentangle graph layer is employed as:

$$\mathbf{e}_{ck}^l = g(\mathbf{e}_{ck}^{l-1}, \{\mathbf{i}, \mathbf{i} \in \mathcal{N}_c\}). \quad (8)$$

The intent-aware representation from different layers is summed to get the final representation:

$$\mathbf{e}_{ck} = \sum_l \mathbf{e}_{ck}^l \quad (9)$$

The graph disentangling module learns the user’s intents distributed in different bundles (global view) from user-item graph. And from bundle-item graph, the graph disentangling module learns the user’s multiple intents within each bundle (local view). Coupling with intents from global and local views, we disentangle the representation of user and bundle, respectively. Specifically, this module disentangles the representation of user and bundle into chunks as $\mathbf{e}_u = (\mathbf{e}_{u1}, \mathbf{e}_{u2}, \dots, \mathbf{e}_{uK})$ and $\mathbf{e}_b = (\mathbf{e}_{b1}, \mathbf{e}_{b2}, \dots, \mathbf{e}_{bK})$.

Cross-View Propagating Module

The user’s representation \mathbf{e}_u and bundle’s representation \mathbf{e}_b couple with the user’s intents from global view and local view, respectively. To exchange intents under different views between the chunks of user and bundle, the LightGCN (He et al. 2020) is employed to the user-bundle graph:

$$\begin{aligned} \mathbf{v}_u &= \sum_{b \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|} \sqrt{|\mathcal{N}_b|}} \mathbf{e}_b, \\ \mathbf{v}_b &= \sum_{u \in \mathcal{N}_b} \frac{1}{\sqrt{|\mathcal{N}_b|} \sqrt{|\mathcal{N}_u|}} \mathbf{e}_u. \end{aligned} \quad (10)$$

Through exchanging views, the chunks \mathbf{v}_u and \mathbf{v}_b represent the user and bundle coupling with intents under local and global view respectively.

Intent Contrasting Module

From the disentangling module and cross-view propagating module, we obtain the user and bundle’s embedding chunks $\mathbf{e}_c = (\mathbf{e}_{c1}, \dots, \mathbf{e}_{cK})$ and $\mathbf{v}_c = (\mathbf{v}_{c1}, \dots, \mathbf{v}_{cK})$ coupling with intents from global and local view, respectively. In the **global view**, more items are involved to present each intent, which can **demonstrates user’s preference** under each intent more clearly. The local view can **reveal the association among items** under the user’s each intent. By contrasting, intents from different views complement each other and can better **present the user’s preference as well as items’ associations** at a finer grain of the user’s intent. Hence we introduce another module which applies InfoNCE(Oord, Li, and Vinyals 2018) to contrast the user and bundle’s embedding capturing intents from different views:

$$L_{contrast} = -\log\left(\frac{\exp(\mathbf{e}_{ck} \cdot \mathbf{v}_{ck+})}{\sum_{k'} \exp(\mathbf{e}_{ck} \cdot \mathbf{v}_{ck'})}\right), \quad (11)$$

where for each chunk of user/bundle’s embedding \mathbf{e}_{ck} , the positive sample \mathbf{v}_{ck+} is the chunk of the same intent within embedding from the different view, while all other chunks are the negative samples.

Prediction and Optimization

Having obtained the presentation of the user and bundle from both global and local views, we concatenate representations from two views to get the final representation and use inner product to estimate the likelihood of their interactions, $\hat{\mathbf{y}}_{ub}$ as:

$$\hat{\mathbf{y}}_{ub} = (\mathbf{e}_u \oplus \mathbf{v}_u) \odot (\mathbf{e}_b \oplus \mathbf{v}_b), \quad (12)$$

where \oplus denotes the concatenation and \odot denotes dot product. Then for model optimization, we adopt the Bayesian Personalized Ranking loss (Rendle et al. 2012):

$$L_{pred} = \sum_{(u,b,d) \in Q} -\ln \sigma(\hat{\mathbf{y}}_{ub} - \hat{\mathbf{y}}_{ud}) + \lambda \cdot \|\theta\|^2, \quad (13)$$

where $Q = \{(u, b, d) | (u, b) \in y^+, (u, d) \in y^-\}$ denotes the training data that involves observed interaction y^+ and unobserved interaction y^- . To avoid over-fitting, we adopt L_2 regularization on the model’s parameter θ which is controlled by the coefficient λ . During training, we alternatively optimize the prediction loss in eq. (12) and contrast loss in eq. (11).

Experiments

To answer the following questions, we conduct experiments on two public datasets.

- **RQ1:** How does MIDGN perform compared with previous approaches?
- **RQ2:** How do different components (Graph Disentangling module in global view, Graph Disentangling module in local view, intent contrast module) affect the results of MIDGN?
- **RQ3:** How do parameters (layer number of Graph Disentangling module, number of intents) influence the results of MIDGN?

Dataset	NetEase	Youshu
User	18,528	8,039
Bundle	22,864	4,771
Item	123,628	32,770
User-Bundle	302,303	51,337
Bundle-item	1,778,838	176,667
User-item	1,128,065	138,515
User-bundle density	0.07%	0.13%
Bundle-item density	0.06%	0.11%
User-item density	0.05%	0.05%

Table 1: Statistics of two utilized datasets

Datasets and metrics Two datasets are used to evaluate our proposed method. The one is NetEase provided by the work(Cao et al. 2017). NetEase is constructed by crawling data from Netease Cloud Music, which enables users to select preferred user-generated playlists or individual songs. Within NetEase, each bundle contains at least 10 songs appearing at least in 5 bundles. Each user consumes at least 10 songs and 10 playlists. The other dataset is named Youshu provided by the work (Chen et al. 2019). Youshu is constructed by crawling data from Youshu, a Chinese book review site. Youshu contains the user’s preference for both individual books and bundles of books. The statistic of these two datasets is shown in Table 1.

We select Recall@K and NDCG@K as evaluation metrics to judge the ranking list performance. Recall@K means the ratio of test bundles within the top-K ranking list. NDCG@K assigning higher scores to the hits at a higher position on the ranking list.

Baselines To demonstrate the effectiveness of the proposed MIDGN, state-of-art methods are chosen for comparison. Specifically, we first chose two matrix factorization based methods, and several graph based methods as follows:

- **MFBPR** (Rendle et al. 2009): This work applies a Bayesian Personalized Ranking learning framework to the matrix factorization method.
- **DAM** (Chen et al. 2019): This work uses the factorized attention mechanism and multi-task framework to capture bundle-level association and collaborate signals.
- **GCN**(Berg, Kipf, and Welling 2017): The GCN method is applied to the user-bundle-item unified graph to predict the user-bundle relations.
- **GCNBG**(Berg, Kipf, and Welling 2017): The GCN method is applied to the user-bundle graph.
- **NGCF** (Wang et al. 2019): This work uses GCN based method to capture high-order collaborative signals for prediction.
- **NGCFBG** (Wang et al. 2019): The NGCF method is applied to the user-bundle graph for prediction.
- **RGCN** (Schlichtkrull et al. 2018): RGCN is GCN based method developed to deal with the multi-relational graph.
- **BasConv** (Liu et al. 2020): This work devises heterogeneous aggregators to learn the embedding of each node within the user-bundle-item graph.

Method	NetEase						Youshu					
	Metrics@20		Metrics@40		Metrics@80		Metrics@20		Metrics@40		Metrics@80	
	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG
MFBPR	0.0335	0.0181	0.0600	0.0246	0.0948	0.0323	0.1959	0.1117	0.2735	0.1320	0.3710	0.1543
DAM	0.0411	0.0210	0.0690	0.0281	0.1090	0.0372	0.2082	0.1198	0.2890	0.1418	0.3915	0.1658
GCN	0.0378	0.0257	0.0625	0.0255	0.1000	0.0342	0.1982	0.1141	0.2661	0.1322	0.3633	0.1541
NGCF	0.0395	0.0207	0.0646	0.0274	0.1021	0.0359	0.1985	0.1143	0.2658	0.1324	0.3542	0.1524
RGCN	0.0402	0.0204	0.0657	0.0272	0.1051	0.0362	0.2032	0.1175	0.2770	0.1371	0.3804	0.1605
BasConv	0.0384	0.0198	0.0636	0.0266	0.1015	0.0350	0.2119	0.1165	0.2761	0.1343	0.3743	0.1561
BGCN	0.0407	0.0210	0.0670	0.0280	0.1112	0.0378	0.2040	0.1069	0.3017	0.1330	0.4169	0.1595
MIDGN	0.0470	0.0244	0.0786	0.0327	0.1241	0.0429	0.2121	0.1204	0.2978	0.1434	0.3988	0.1548
%Improv.	0.0491	0.0258	0.0829	0.0346	0.1304	0.0453	0.2347	0.1345	0.3248	0.1593	0.4355	0.1851
	0.0678	0.0343	0.1085	0.0451	0.1654	0.0578	0.2682	0.1527	0.3712	0.1808	0.4817	0.2063
	38.2%	33.1%	30.9%	30.3%	26.8%	27.5%	14.2%	13.8%	14.2%	13.6%	10.7%	11.3%

Table 2: Performance comparisons on two real-world datasets

- **BGCN** (Chang et al. 2020): BGCN proposes a graph neural network model to explicitly model complex relations between users, items, and bundles.

The negative sampling is set to 1, the learning rate is selected from 1e-5, 3e-5, 1e-4, 3e-4, 1e-3, 3e-3, and we adopt BPR loss for all methods and employ Adam optimizer with the 4096-size mini-batch and fit the embedding size as 64. We make use of Nvidia Titan RTX graphics card equipped with AMD r9-5900x CPU (32GB Memory). For MIDGN, the number of intents and the number of layers are selected from 1, 2, 4, 8 and 1, 2, 3, 4, respectively

Performance Comparison(RQ1)

The results of all the methods are reported in Table 2. For the results, we have the following observations.

- **Our proposed MIDGN achieves the best results.** MIDGN significantly outperforms all the baselines both on the metrics of Recall@K and NDCG@K. On NetEase, MIDGN improves the performance over the best baseline by 26.8%-38.2%. And MIDGN outperforms the best result by 10.7%-14.2% on Youshu. We contribute the improvement to the following aspects: 1) By grouping the items to model the user’s intents, MIDGN reduces the noise from individual items and thus better learns the user’s preference. 2) Disentangling the embedding of user and bundle according to the user’s intents presents the user and bundle at a more granular level. and 3) By learning and contrasting the user’s intents under global and local views, MIDGN clearly captures the user’s intents.
- **Graph models are effective in personalized bundle recommendation.** Graph models (GCN, NGCF, RGCN) are effective in bundle recommendation, which can be proved by their better performance than MFBPR. We attribute this to their superiority in capturing graph structure and multi-hop collaborative information. Among these models, RGCN performs the best, which denotes

the importance of distinguishing the different relationships for the problem of personalized bundle recommendation.

- **Capturing the user’s preference on items’ associations is important.** Failing to capture bundle-level association, graph models (GCN, NGCF, RGCN) can not even surpass DAM, which employs attention mechanism and multi-tasks framework. By being devised to capture the user’s preference on items’ associations, BasConv and BGCN significantly improve the performance of personalized bundle recommendation and performs the best among baselines.

Study of MIDGN (RQ2&RQ3)

Next we investigate the underlining mechanism of our MIDGN with three ablated models: MIDGN_{w/o contra.} that removes the contrast module, MIDGN_{w/o local} that replaces the graph disentangling module in the local view with GCN, and MIDGN_{w/o global} that removes the graph disentangling module in the global view and uses GCN instead. From the results in 3, we have the following observations:

- MIDGN outperforms MIDGN_{w/o global} significantly. This demonstrate the importance of global view in modeling the user’s multiple intents. And involving more items is helpful to learn the user’s intents.
- MIDGN_{w/o local} is the least competitive. We contribute this to its failure on learning associations between items under the user’s each intent.
- MIDGN outperforms MIDGN_{w/o contra.}. This demonstrates the effectiveness of the intent contrast module in helping the intents from different views complement each other.
- All the ablated models significantly outperform the baselines. This demonstrates the effectiveness of disentangling multi-view intents for bundle recommendation.

Method	NetEase Metrics@20		Youshu Metrics@20	
	Recall	NDCG	Recall	NDCG
MIDGN _{w/o local}	0.0547	0.0287	0.2475	0.1393
MIDGN _{w/o global}	0.0583	0.0303	0.2464	0.1441
MIDGN _{w/o contra.}	0.0609	0.0312	0.2588	0.1471
MIDGN	0.0678	0.0343	0.2682	0.1527

Table 3: Ablated models analysis

Method-L	NetEase Metrics@20		Youshu Metrics@20	
	Recall	NDCG	Recall	NDCG
MIDGN-1	0.0442	0.0233	0.2348	0.1348
MIDGN-2	0.0559	0.0295	0.2503	0.1429
MIDGN-3	0.0642	0.0331	0.2682	0.1527
MIDGN-4	0.0678	0.0343	0.2617	0.1472

Table 4: Impact of Layer Number(L)

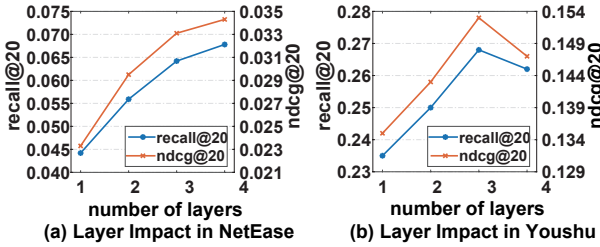


Figure 3: Impact of Layer Number(L)

We also conduct parameter studies to further investigate the influence of some parameters. Specifically, how the number of disentangling layers, the number of the user’s intents influence the performance of MIDGN.

Impact of Layer Number The graph disentangling layer disentangles the user’s intents and collects groups of items pertinent to individual intents to learn intent-aware representations. By stacking more layers, collaborative information from multi-hop neighbors is distilled. We investigate how the number of layers influences the performance of MIDGN. Specifically, we conduct experiments with layer L in range $\{1, 2, 3, 4\}$ and the results are shown in Table 4, where MIDGN-L is to illustrate that L layers are involved in disentangling module. There are some observations:

- Increasing the number of layers can improve the performance of our model. MIDGN-2 highly outperforms MIDGN-1. The reason is MIDGN-1 only gains information from one-hop neighbor and neglects high-order collaborative information.
- When increasing the layer of number, the performance does not always improve. MIDGN-3 outperforms MIDGN-4 on data Youshu. This can be attributed to the noise which increases along with the hop of neighbor.

Impact of Intent Number To investigate the impact of intent number, we search the number K of Intent in range

Method-K	NetEase Metrics@20		Youshu Metrics@20	
	Recall	NDCG	Recall	NDCG
MIDGN-1	0.0579	0.0299	0.2523	0.1442
MIDGN-2	0.0652	0.0340	0.2632	0.1497
MIDGN-4	0.0678	0.0343	0.2682	0.1527
MIDGN-8	0.0587	0.0302	0.2517	0.1439

Table 5: Impact of Intent Number(K)

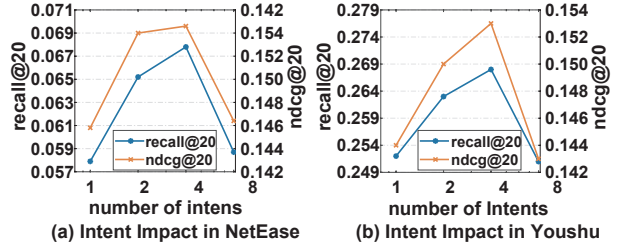


Figure 4: Impact of Intent Number(K)

$\{1, 2, 4, 8\}$. From the results in Table 5, we have the following observations:

- MIDGN performs the worst when the intent number $K = 1$. This illustrates the diversity of the user’s intents and an unitary intent can not present the user’s preference effectively.
- When the intent number K increases from 4 to 8, the performance drops sharply. This suggests the model suffers from too fine-grained intents. The reason may be that when cluster items into too many groups for individual intents, each group can not present the intent coupling with it well.

Conclusion

In this paper, we explore the diversity of the user’s intents in the problem of personalized bundle recommendation. We propose a novel model named Multi-view Intent Disentangle Graph Networks(MIDGN) which disentangles the user’s intents from both the global and local views. With the help of GNN equipped with the neighbor routing mechanism, MIDGN disentangles user-item and bundle-item graph coupling with the user’s intents from global and local views, respectively. Meanwhile, MIDGN compares the intents learned from different views to better represent the user’s preference as well as the items’ associations at a finer grain of the user’s intent. Extensive experiments on Youshu and NetEase demonstrate the superiority of the proposed MIDGN.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant No.61602197, Grant No.L1924068, Grant No.61772076, in part by CCF-AFSG Research Fund under Grant No.RF20210005, and in part by the fund of Joint Laboratory of HUST and Pingan Property

& Casualty Research (HPL). The authors would also like to thank the anonymous reviewers for their comments on improving the quality of this paper.

References

- Bai, J.; Zhou, C.; Song, J.; Qu, X.; An, W.; Li, Z.; and Gao, J. 2019. Personalized bundle list recommendation. In *The World Wide Web Conference*, 60–71.
- Berg, R. v. d.; Kipf, T. N.; and Welling, M. 2017. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*.
- Cao, D.; Nie, L.; He, X.; Wei, X.; Zhu, S.; and Chua, T.-S. 2017. Embedding factorization models for jointly recommending items and user generated lists. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 585–594.
- Chang, J.; Gao, C.; He, X.; Jin, D.; and Li, Y. 2020. Bundle recommendation with graph convolutional networks. In *Proceedings of the 43rd international ACM SIGIR conference on Research and development in Information Retrieval*, 1673–1676.
- Chen, L.; Liu, Y.; He, X.; Gao, L.; and Zheng, Z. 2019. Matching User with Item Set: Collaborative Bundle Recommendation with Deep Attention Network. In *IJCAI*, 2095–2101.
- Chen, Y.-L.; Tang, K.; Shen, R.-J.; and Hu, Y.-H. 2005. Market basket analysis in a multiple store environment. *Decision support systems*, 40(2): 339–354.
- Garfinkel, R.; Gopal, R.; Tripathi, A.; and Yin, F. 2006. Design of a shopbot and recommender system for bundle purchases. *Decision Support Systems*, 42(3): 1974–1986.
- He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; and Wang, M. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 639–648.
- Kipf, T. N.; and Welling, M. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.
- Le, D. T.; Lauw, H. W.; and Fang, Y. 2017. Basket-sensitive personalized item recommendation. *IJCAI*.
- Liu, G.; Fu, Y.; Chen, G.; Xiong, H.; and Chen, C. 2017. Modeling buying motives for personalized product bundle recommendation. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 11(3): 1–26.
- Liu, Q.; Ge, Y.; Li, Z.; Chen, E.; and Xiong, H. 2011. Personalized travel package recommendation. In *2011 IEEE 11th International Conference on Data Mining*, 407–416. IEEE.
- Liu, Y.; Xie, M.; and Lakshmanan, L. V. 2014. Recommending user generated item lists. In *Proceedings of the 8th ACM Conference on Recommender systems*, 185–192.
- Liu, Z.; Wan, M.; Guo, S.; Achan, K.; and Yu, P. S. 2020. Basconv: Aggregating heterogeneous interactions for basket recommendation with graph convolutional neural network. In *Proceedings of the 2020 SIAM International Conference on Data Mining*, 64–72. SIAM.
- Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.
- Parameswaran, A.; Venetis, P.; and Garcia-Molina, H. 2011. Recommendation systems with complex constraints: A course recommendation perspective. *ACM Transactions on Information Systems (TOIS)*, 29(4): 1–33.
- Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, 452–461.
- Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*.
- Rendle, S.; Freudenthaler, C.; Gantner, Z.; and Schmidt-Thieme, L. B. 2014. Bayesian personalized ranking from implicit feedback. In *Proc. of Uncertainty in Artificial Intelligence*, 452–461.
- Sar Shalom, O.; Koenigstein, N.; Paquet, U.; and Vanchinathan, H. P. 2016. Beyond collaborative filtering: The list recommendation problem. In *Proceedings of the 25th international conference on world wide web*, 63–72.
- Schlichtkrull, M.; Kipf, T. N.; Bloem, P.; Van Den Berg, R.; Titov, I.; and Welling, M. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*, 593–607. Springer.
- Wang, X.; He, X.; Wang, M.; Feng, F.; and Chua, T.-S. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, 165–174.
- Wang, X.; Jin, H.; Zhang, A.; He, X.; Xu, T.; and Chua, T.-S. 2020a. Disentangled graph collaborative filtering. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1001–1010.
- Wang, Y.; Tang, S.; Lei, Y.; Song, W.; Wang, S.; and Zhang, M. 2020b. DisenHAN: Disentangled Heterogeneous Graph Attention Network for Recommendation. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 1605–1614.
- Wang, Z.; Wei, W.; Cong, G.; Li, X.-L.; Mao, X.-L.; and Qiu, M. 2020c. Global context enhanced graph neural networks for session-based recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 169–178.
- Wang, Z.; Wei, W.; Cong, G.; Li, X.-L.; Mao, X.-L.; Qiu, M.; and Feng, S. 2020d. Exploring Global Information for Session-based Recommendation. *arXiv preprint arXiv:2011.10173*.
- Wang, Z.; Wei, W.; Cong, G.; Mao, X.-L.; Li, X.-L.; and Feng, S. 2020e. Exploiting Repeated Behavior Pattern and Long-term Item dependency for Session-based Recommendation. *arXiv preprint arXiv:2012.05422*.
- Wei, W.; Liu, J.; Mao, X.; Guo, G.; Zhu, F.; Zhou, P.; and Hu, Y. 2019. Emotion-aware chat machine: Automatic emotional response generation for human-like emotional inter-

action. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 1401–1410.

Wu, S.; Tang, Y.; Zhu, Y.; Wang, L.; Xie, X.; and Tan, T. 2019. Session-based recommendation with graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 346–353.

Xie, M.; Lakshmanan, L. V.; and Wood, P. T. 2010. Breaking out of the box of recommendations: from items to packages. In *Proceedings of the fourth ACM conference on Recommender systems*, 151–158.

Xie, M.; Lakshmanan, L. V.; and Wood, P. T. 2014. Generating top-k packages via preference elicitation. *Proceedings of the VLDB Endowment*, 7(14): 1941–1952.

Zhao, S.; Zhang, Y.; Wang, S.; Zhou, B.; and Cheng, C. 2019. A recurrent neural network approach for remaining useful life prediction utilizing a novel trend features construction method. *Measurement*, 146: 279–288.

Zheng, V.; Cao, B.; Zheng, Y.; Xie, X.; and Yang, Q. 2010. Collaborative filtering meets mobile recommendation: A user-centered approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24.

Zhu, T.; Harrington, P.; Li, J.; and Tang, L. 2014. Bundle recommendation in ecommerce. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, 657–666.