Solving the Sparsity Problem in Recommendations via Cross-Domain Item Embedding Based on Co-Clustering

Yaqing Wang[1], Chunyan Feng[1,2], Caili Guo[1,2], Yunfei Chu[1,2] and Jenq-Neng Hwang[3]
[1]Beijing Key Laboratory of Network System Architecture and Convergence,
School of Information and Communication Engineering,
Beijing University of Posts and Telecommunications, Beijing, China
[2]Beijing Laboratory of Advanced Information Networks, Beijing, China
[3]Department of Electrical Engineering, University of Washington, Seattle, USA
{wangyq,cyfeng,guocaili,yfchu}@bupt.edu.cn,hwang@uw.edu

ABSTRACT

Session-based recommendations recently receive much attentions due to no available user data in many cases, e.g., users are not logged-in/tracked. Most session-based methods focus on exploring abundant historical records of anonymous users but ignoring the sparsity problem, where historical data are lacking or are insufficient for items in sessions. In fact, as users' behavior is relevant across domains, information from different domains is correlative, e.g., a user tends to watch related movies in a movie domain, after listening to some movie-themed songs in a music domain (i.e., cross-domain sessions). Therefore, we can learn a complete item description to solve the sparsity problem using complementary information from related domains. In this paper, we propose an innovative method, called **C**ross-**D**omain **I**tem **E**mbedding method based on **C**o-clustering (CDIE-C), to learn cross-domain comprehensive representations of items by collectively leveraging single-domain and cross-domain sessions within a unified framework. We first extract cluster-level correlations across domains using co-clustering and filter out noise. Then, cross-domain items and clusters are embedded into a unified space by jointly capturing item-level sequence information and cluster-level correlative information. Besides, CDIE-C enhances information exchange across domains utilizing three types of relations (i.e., item-to-context-item, item-to-context-co-cluster and co-cluster-to-context-item relations). Finally, we train CDIE-C with two efficient training strategies, i.e., joint training and two-stage training. Empirical results show CDIE-C outperforms the state-of-the-art recommendation methods on three cross-domain datasets and can effectively alleviate the sparsity problem.

KEYWORDS

Cross-domain recommendations; the sparsity problem; item embedding; co-clustering

ACM Reference Format:

Yaqing Wang, Chunyan Feng, Caili Guo, Yunfei Chu and Jenq- Neng Hwang. 2019. Solving the Sparsity Problem in Recommendations via Cross-Domain Item Embedding Based on Co-clustering. In The Twelfth ACM International Conference on Web Search and Data Mining (WSDM '19), February 11–15, 2019, Melbourne, VIC, Australia. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3289600.3290973

WSDM '19, February 11–15, 2019, Melbourne, VIC, Australia

ACM ISBN 978-1-4503-5940-5/19/10…$15.00
https://doi.org/10.1145/3289600.3290973

1 INTRODUCTION

Most recommender systems utilize user history records to help users faced with overwhelming selections of items by identifying particular items that are likely to match each user's preferences [3]. However, such user-based recommendation models are often not available when the user is not logged in or not tracked for either technical or privacy reasons [11]. Thus, in recent years, many researchers have begun to pay attention to session-based recommendations [1, 11, 12, 14, 20, 24, 25] where detailed user information is not required, instead only the behavior of users is often considered.

Most session-based recommender systems are based on either relatively simple methods (e.g., item-to-item similarity [20, 24]) or information of whole sessions (i.e., single-domain sessions) [1, 11, 12, 14, 25]. While effective, those methods often take into account items with abundant interactions, with little attention paid to sparse systems where available historical logs are lacking or are insufficient since even popular items can co-occur with only a limited number of items in a system. Thus, the utility of such methods become degraded due to the sparsity problem. On the other hand, as the variety of web services has increased significantly, complementary information about the sparse system can be obtained from other domains [28]. For examples, users will buy movie-related products in e-commerce domains or listen to movie-themed songs in music domains after watching a movie in movie domains, and vice versa. The movie and related songs/products of different domains with their temporal information compose cross-domain sessions. Therefore, an interesting question arises—how do we fuse correlative information of multiple session-based domains to solve the sparsity problem?

Cross-domain recommender systems [4], which fuse correlative information of different domains to improve the performance of recommendations, have gained growing research attention in recent years. Due to privacy protection and the difference of domains, cross-domain user information and item information are almost non-overlapping. Some studies in the absence of user/item overlaps link multiple domains by using auxiliary information such as external knowledge repositories like DBpedia [7, 15]. Nevertheless, the performance of these methods depends on the quality of the auxiliary information. Besides, many collaborative filtering methods are proposed to transfer cross-domain knowledge by sharing cross-domain common patterns [9, 13, 18], which are hard to apply in session-based recommendations for multiple domains due to the absence of user profiles. We are thus motivated to learn cross-domain comprehensive item descriptions to solve the sparsity problem by fusing sessions from different domains (i.e., single-domain/cross-domain sessions) in a unified way.

It is, however, a challenging task to fuse the multi-domain information because of the following two main issues. Firstly, the information is noisy across different domains. For example, due to popularity or necessity, many items often appear in the same sessions with others. Nevertheless, they are not actually related to each other. These items will make it difficult to obtain the real correlative information between items due to the introduced noise. Secondly, information of single-domain sessions and cross-domain sessions is rather diverse. Specifically, single-domain sessions are composed of homogeneous relations formed by items in a single domain [20] and sequence information [12]. Meanwhile, cross-domain sessions include cross-domain items (such as movies and songs), which form heterogeneous relations. The homogeneous/heterogeneous relations and sequence information come from different feature spaces, which makes it hard to fuse them.

In this paper, we propose an innovative method, called cross-domain item embedding method based on co-clustering (CDIE-C), to learn the comprehensive representations of items through jointly leveraging single-domain sessions and cross-domain sessions in a unified framework. First, we extract the cluster-level correlations by exploring cross-domain co-occurrence relations of items (i.e., cross-domain sessions) based on the co-clustering method and filter out the noise. Second, we capture item-level sequence information (i.e., single-domain sessions) and cluster-level correlation information using three types of relations (i.e., item-to-context-item, item-to-context-co-cluster and co-cluster-to-context-item relations). Third, we learn effective item representations by fusing three types of relations into a unified low-dimensional space with two training strategies, i.e., joint training and two-stage training.

The main contributions of this work are summarized as follows.

(1) We focus on learning comprehensive representations of items by jointly leveraging single-domain sessions and cross-domain sessions in a unified way; and alleviate the sparsity problem by fusing complementary information from different domains.
(2) We propose a novel embedding method CDIE-C, which embeds item-level sequence information and cluster-level correlative information from different domains into a unified low-dimensional space. Furthermore, the model is trained in two efficient ways, i.e., joint training and two-stage training.
(3) We demonstrate the performance of CDIE-C on three cross-domain datasets. A series of experimental results validate that our proposed CDIE-C can achieve better performance than the state-of-the-art methods and can work well even with sparse item information.

The remainder of the paper is organized as follows. Section 2 reviews existing work related to our research. Section 3 details the techniques of our proposed CDIE-C method. Experimental results on three cross-domain datasets are presented in Section 4, followed by the conclusion of our work in Section 5.

2 RELATED WORK

2.1 Session-based Recommendations

There are many studies for session-based recommendations that rely solely on anonymous users' actions in an ongoing single-domain session, such as item-to-item similarity [20, 24], item sequences [1, 11, 12, 14, 25].

The most natural solution for session-based recommendations is the item-to-item recommendation[20, 24] with a simplified setting, which precomputes an item-to-item similarity matrix from available data. While effective, these methods often only take into account the similarity of items, in effect losing the order information.

Another type of methods focuses on information of the whole session. More specifically, RNNs-based methods are commonly applied to capture sequential signals [12, 14, 25]. Balázs et al., who proposed GRU4REC [12], were among the first to explore RNNs for item sequences. Tan et al. in [25] proposed an enhanced version of GRU4REC, which is trained on embedded sequences with random drop-out to reduce overfitting. Jannach et al. [14] combine the kNN approach with GRU4REC to leverage sequential signals and the co-occurrence information jointly. Besides, as word embedding (e.g., word2vec [22]) has succeeded in capturing fine-grained semantic and syntactic regularities, many session-based recommendation methods [1, 11] transfer a session of items into a sentence of words for exploiting sequence information.

The above researches aim to explore abundant history records in sessions, in fact ignoring the sparsity problem, which can be effectively solved by our introducing complementary information from related domains using cross-domain correlations.

2.2 Cross-domain Recommendations

Due to privacy or difference of domains reasons, cross-domain user and item information are almost non-overlapping. Therefore, we focus on existing cross-domain methods in the absence of user/item overlaps and group them into the following two main categories depending on the availability of auxiliary information.

The first category of methods transfers cross-domain information by using the auxiliary information (such as DBpedia). Fernndez-Tobas et al. [7] and Kaminskas et al. [15] proposed a knowledge-based framework built on the DBpedia ontology, which is used to measure the relatedness between items in different domains. The performance of their methods is degraded when items cannot be defined based on the knowledge repository due to no available data. The second category of methods aims to establish a bridge between two domains based on the cluster-level user-item rating patterns. Li et al. [18] first proposed the rating patterns sharing approach, which can transfer knowledge from the source domain to the target domain. Gao et al. [9] proposed Cluster-level Latent Factor Model (CLFM) to partition the rating patterns across domains into common and domain-specific parts, under the assumption that two domains have the same rating patterns is unrealistic in practice. Due to the design based on user-item interactions, these methods are not suitable for the session-based recommendations with unknown user information. Thus, we aim to fuse cross-domain correlative

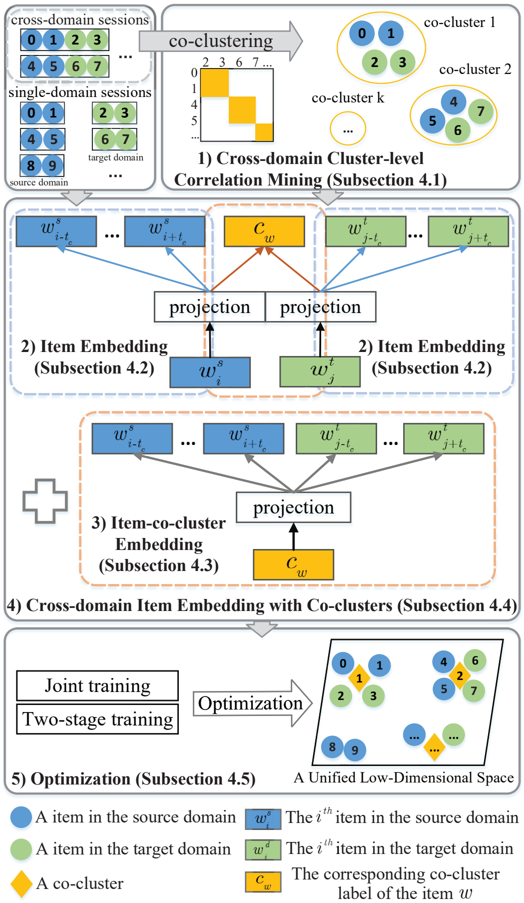


Figure 1: Illustration of the CDIE-C framework.

information based on item sequences in different domains (without user information) to solve the sparsity problem.

3 PROPOSED METHOD

In this section, we present our CDIE-C method to learn comprehensive representations of items by jointly utilizing single-domain sessions and cross-domain sessions. The illustration of CDIE-C is depicted in Figure 1. We first explore cross-domain cluster-level correlations with cross-domain sessions and filter out the noise (Subsection 3.1). Then, we present a unified model (Subsection 4.4) mathematically based on item-level sequence information (i.e., single-domain and cross-domain sessions) and cluster-level correlation information in terms of item embedding (Subsection 4.2) and item-co-cluster embedding (Subsection 4.3). Finally, we introduce the joint learning and two-stage learning strategies (Subsection 4.5) for learning item representations and co-cluster representations of different domains in a unified low-dimensional space.

3.1 Cross-domain Cluster-level Correlation Mining

As discussed in Section 1, users' behavior is relatively consistent in similar situations [8]. That is, they will pay attention to similar items in different domains, such as movie-themed songs and related movies. Actually, these similar items constitute cross-domain cluster-level correlations. Besides, it is a hard task to exploit the correlations, as cross-domain information is noisy. In this paper, we adopt the co-clustering method [6] to explore the correlations and filter noise for fusing complementary information between different domains.

First, we introduce a bipartite graph model to represent co-occurrence relations of items across domains. An undirected bipartite graph is a triplet $G = \{S, T, E\}$, where $S = \{w_1^s, w_2^s, ..., w_{I^s}^s\}$ and $T = \{w_1^t, w_2^t, ..., w_{I^t}^t\}$ are two sets of items of the source domain s and the target domain t; I^s and I^t are the number of items in the domains s and t, respectively; $E = S \times T$ is the set of edges in G, and $e_{ij} \in E$ represents co-occurrence times of s_i and t_j in all sessions. We establish the adjacent matrix of cross-domain items $A \in \Re^{I^s \times I^t}$ according to the bipartite graph G, where A_{ij} is co-occurrence times of item i in the source domain s and item j in the target domain t.

To improve the efficiency of computation, we decompose A based on SVD as follows:

$$A_n = U \Sigma V^{\mathrm{T}} = D_1^{-1/2} A D_2^{-1/2}, \tag{1}$$

where D_1 and D_2 are the diagonal matrices with $D_1(i,i) = \sum_{j=1}^{J} A_{ij}$ and $D_2(j,j) = \sum_{i=1}^{I} A_{ij}$, respectively.

The desired partitioning information is provided by the first $\ell = \lceil \log_2^k \rceil$ singular vectors, starting from the second vector. The ℓ singular vectors are then used to form the matrix Z:

$$Z = \begin{bmatrix} D_1^{-1/2} U \\ D_2^{-1/2} V \end{bmatrix}, \tag{2}$$

where the columns of U are the ℓ singular vectors $u_2, \ldots, u_{\ell+1}$, and similarly for V.

Finally, we can obtain the set of co-clusters $C^a = \{c_1, c_2, \ldots c_l\}$ by clustering the row of Z with k-means. In our experiments, we notice that some co-clusters are noisy in capturing cross-domain correlations. Some items appear frequently because of their popularity, but they are not actually related to other items. It will make some co-clusters too large due to noise. Therefore, instead of taking into account the complete set of co-clusters, we select the set $C = \{c_k \| c_k| / (I^s + I^t) < p, c_k \in C^a\}$, where p is the correlation threshold and $|c_k|$ is the number of items in c_k. By setting a suitable threshold p, we can improve not only the computational efficiency with a reduced number of co-clusters but also the robustness by filtering out the noise.

3.2 Item Embedding

Sequences of items are analogous to sequences of words [5, 19], so we can use word embedding models to learn item representations [1, 10]. Given sequences of items in single-domain sessions, we first apply item embedding to capture item-level sequence information. Specifically, given sequences of items $D = (w_i)_{i=1}^{K}$ from a finite item set $F = \{w_i\}_{i=1}^{I}$, the skip-gram objective aims at maximizing the following term:

$$L_{item-S} = \frac{1}{K} \sum_{i=1}^{K} \sum_{-t_c \le j \le t_c, j \ne 0} \log p(w_{i+j} | w_i), \tag{3}$$

where t_c denotes the context window size. Obviously, the target form is consistent with the skip-gram word2vec model.

Levy and Goldberg [17] demonstrated that the skip-gram with negative-sampling word2vec model implicitly factorizes the point-wise mutual information (PMI) matrix shifted by a global constant. PMI between a word i and its context j is defined as,

$$PMI(i,j) = \frac{\#(i,j)\cdot \mathcal{D}}{\#(i)\cdot \#(j)}, \tag{4}$$

where $\#(i) = \sum_j \#(i,j)$, $\#(j) = \sum_i \#(i,j)$ and $\mathcal{D} = \sum_{ij} \#(i,j)$. Levy and Goldberg [17] further proposed to perform the word embedding by the spectral dimensionality reduction (e.g., singular value decomposition) on the shifted positive PMI (SPPMI) matrix,

$$SPPMI(i,j) = max\{PMI(i,j) - \log h, 0\}, \tag{5}$$

where h is a hyperparameter that controls the sparsity of the SPPMI matrix. The model can be easily trained on count-aggregated data (i.e., $(i, j, \#(i, j))$ triplets), making it more applicable to much larger datasets than the skip-gram word2vec model's training procedure [17]. Thus, we will follow similar idea in our method.

Let $X \in \Re^{I\times J}$ be the SPPMI matrix, where I denotes the number of items in sessions and J represents the number of context items[1]. We divide sessions into shorter segments according to the context window size t_c. $X_{ij} \in X$ can now be computed according to Eq. (4) and Eq. (5). Due to the good performance of Bayesian Personalized Ranking (BPR) [23] in session-based recommendations [5, 12], we construct our objective function by following BPR. The optimization criterion for BPR contains a Bayesian analysis and the prior probability for the model parameters. Then, we can derive the objective function for item embedding by formulating a maximum posterior estimator:

$$\begin{aligned}
\mathcal{L}'_{item} &= \ln p(\boldsymbol{\theta}|>_i) \\
&= \ln p(>_i|\boldsymbol{\theta})p(\boldsymbol{\theta}) \\
&= \ln \prod_{(i,j,k)\in\mathcal{D}_x} \sigma(\hat{x}_{ijk})p(\boldsymbol{\theta}) \\
&= \sum_{(i,j,k)\in\mathcal{D}_x} \ln\sigma(\hat{x}_{ijk}) + \ln p(\boldsymbol{\theta}) \\
&= \sum_{(i,j,k)\in\mathcal{D}_x} \ln\sigma(\hat{x}_{ijk}) - \lambda_{\boldsymbol{\theta}}\|\boldsymbol{\theta}\|^2,
\end{aligned} \tag{6}$$

where $>_i$ is the desired latent preference structure for item w_i; the observed subset $\mathcal{D}_x$ of $>_i$ is used as training data; the model parameters for matrix factorization are $\boldsymbol{\theta} = (P, Q)$; $P \in \Re^{I\times d}$ and $Q \in \Re^{J\times d}$ are d-dimension item and context representations; $x_{ijk} = x_{ij} - x_{ik}$, $x_{ij} = p_i^{\mathrm{T}} q_j$ and $x_{ik} = p_i^{\mathrm{T}} q_k$; $p_i \in P$ and $q_j \in Q$ denote d-dimension representations of item i and context j, respectively; $\sigma = \frac{1}{1+e^{-x}}$ is the logistic sigmoid function and $\lambda_{\boldsymbol{\theta}}$ is the model specific regularization parameters.

Finally, we define SPPMI matrices $X^s \in \Re^{I^s\times J^s}$ and $X^t \in \Re^{I^t\times J^t}$ for the source domain s and the target domain t. Based on the objective function Eq.(6), we can complete item embedding for two domains s and t by maximizing the following objective function:

$$\mathcal{L}_{item} = \sum_{(i,j,k)\in\mathcal{D}_x^s} \ln\sigma(\hat{x}_{ijk}^s) + \sum_{(i,j,k)\in\mathcal{D}_x^t} \ln\sigma(\hat{x}_{ijk}^t) - \lambda_{\boldsymbol{\theta}}\|\boldsymbol{\theta}\|^2, \tag{7}$$

where $\mathcal{D}_x^s$ and $\mathcal{D}_x^t$ are the training corpus for the domains s and t and $\lambda_{\boldsymbol{\theta}}$ is the model specific regularization parameter for $\boldsymbol{\theta} = (P^s, P^t, Q^s, Q^t)$.

3.3 Item-co-cluster Embedding

In this section, we first propose a new embedding model for heterogenous relations, i.e., item-cluster embedding. Item embedding is used to predict the surrounding items (i.e., context items) within a certain distance based on the current one. Similar to the idea, the item-cluster embedding aims at predicting the corresponding cluster labels given the current item (i.e., item-to-context-cluster relations) and the surrounding context items given the corresponding cluster labels of the current item (i.e., cluster-to-context-item relations) jointly. Both item-to-context-cluster and cluster-to-context-item relations describe cluster-level correlations. The difference is that the former encodes the dependency relationship of the item to its related cluster labels, while the latter extends impacts of the correlations by describing the relationship between the cluster labels of the current item and context items. This method can be applied to heterogeneous cases, which is an extension of item embedding. Specifically, given an item set $F = \{w_i\}_{i=1}^I$ with sequences of items D and their cluster labels $C'_l = (C'_{w_i})_{i=1}^I$, where each item can simultaneously belong to several clusters (i.e. $C'_{w_i} = \{c'_{w_i,1}, c'_{w_i,2}\ldots\}$[2]), the objective aims at maximizing the following terms:

$$\begin{aligned}
\mathcal{L}_{cls-S} =& \frac{1}{I}\sum_{i=1}^{I}\sum_k \log p(c'_{w_i,k}|w_i) \\
&+ \frac{1}{|C'_l|}\sum_{i,k}\sum_{-t_c\le j\le t_c, j\ne 0} \log p(w_{i+j}|c'_{w_i,k}),
\end{aligned} \tag{8}$$

where $|C'_l|$ is the number of cluster labels for the item set F; the first and second terms of the objective function denote the item-cluster embedding in terms of item-to-context-cluster relations and cluster-to-context-item relations, respectively.

Then, based on cross-domain co-clusters extracted by Section 2.1, we aim to perform item-co-cluster embedding in multiple domains. Given two item sets $F^s = \{w_i^s\}_{i=1}^{I^s}$, $F^t = \{w_i^t\}_{i=1}^{I^t}$ and their corresponding co-cluster labels $C_l^s = (C_{w_i^s}^s)_{i=1}^{I^s}$ and $C_l^t = (C_{w_i^t}^t)_{i=1}^{I^t}$, the item-co-cluster embedding only extends the collection of items, when compared with the item-cluster embedding in the single domain. Thus, we maximize the objective function for items and co-clusters as follows:

$$\begin{aligned}
\mathcal{L}_{co-S} =& \sum_{r\in\{s,t\}}\frac{1}{I^r}\sum_{i=1}^{I^r}\sum_k \log p(c^r_{w_i,k}|w_i^r) \\
&+ \sum_{r\in\{s,t\}}\frac{1}{|C_l^r|}\sum_{i,k}\sum_{-t_c\le j\le t_c, j\ne 0} \log p(w^r_{i+j}|c^r_{w_i,k}),
\end{aligned} \tag{9}$$

where r indicates different domains, i.e., the target domain s or the source domain t.

[1] In experiments, the set of items in sessions and the set of context items are the same.

[2] Each item belongs to only a cluster in this paper.

As the item-co-cluster embedding model is an extension to item embedding, the equivalent matrix factorization for item embedding can naturally expand to the embedding model for items and co-clusters. Therefore, we define item and context co-cluster SPPMI matrices as $Y^s \in \Re^{I^s \times L}$ and $Y^t \in \Re^{I^t \times L}$, and co-cluster and context item SPPMI matrices as $Z^s \in \Re^{L \times J^s}$ and $Z^t \in \Re^{L \times J^t}$, where $I^{s/t}$ and $J^{s/t}$ are the number of items and context items in the domain s and the domain t, respectively, and L is the number of co-clusters shared by the two domains. The elements in Y^s, Y^t, Z^s and Z^t are computed according to Eq. (4) and Eq. (5). In Figure 2, we show the relationships between SPPMI matrices $X^{s/t}$, $Y^{s/t}$ and $Z^{s/t}$ in two domains, and I_d is an identity matrix. Finally, item-co-cluster embedding can be realized based on the objective function (6) by maximizing the following objective function:

$$\begin{aligned}\mathcal{L}_{co} = &\sum_{r \in \{s,t\}} \sum_{(i,j,k) \in \mathcal{D}_y^r} \ln \sigma(\hat{y}_{ijk}^r) \\ &+ \sum_{r \in \{s,t\}} \sum_{(i,j,k) \in \mathcal{D}_z^r} \ln \sigma(\hat{z}_{ijk}^r) - \lambda_{\theta} \|\theta\|^2,\end{aligned} \tag{10}$$

where $\mathcal{D}_y^r$ is the training corpus for item-to-context-co-cluster entry $\hat{y}_{ij}^r$; $\mathcal{D}_z^r$ is the training corpus for co-cluster to context item entry $\hat{z}_{ij}^r$; θ denotes the model parameter $(P^s, P^t, Q^s, Q^t, R, O)$; and $R \in \Re^{L \times d}$ and $O \in \Re^{L \times d}$ are co-cluster and context co-cluster representations, respectively.

3.4 Cross-domain Item Embedding with Co-clusters

Both the item embedding and item-co-cluster embedding models infer latent item representations. The difference is that the item representations inferred from item embedding focus on sequence information in single domains, while those inferred from the item-co-cluster embedding encode cluster-based correlations between items in different domains. We propose learning these representations jointly:

$$\begin{aligned}\mathcal{L} = &\sum_{(i,j,k) \in \mathcal{D}_x^s} \ln \sigma(\hat{x}_{ijk}^s) + \sum_{(i,j,k) \in \mathcal{D}_x^t} \ln \sigma(\hat{x}_{ijk}^t) \\ &+ \lambda_c \sum_{(i,j,k) \in \mathcal{D}_y^s} \ln \sigma(\hat{y}_{ijk}^s) + \lambda_c \sum_{(i,j,k) \in \mathcal{D}_y^t} \ln \sigma(\hat{y}_{ijk}^t) \\ &+ \lambda_c \sum_{(i,j,k) \in \mathcal{D}_z^s} \ln \sigma(\hat{z}_{ijk}^s) + \lambda_c \sum_{(i,j,k) \in \mathcal{D}_z^t} \ln \sigma(\hat{z}_{ijk}^t) \\ &- \lambda_{\theta} \|\theta\|^2,\end{aligned} \tag{11}$$

where the set θ denotes all parameters $(P^s, P^t, Q^s, Q^t, R, O)$; λ_{θ} is the model specific regularization parameter; and λ_c balances the performance between the item embedding and the item-co-cluster embedding. Setting its relative scale smaller will impose more regularization from the item embedding and vice versa. In our empirical study, we will select these hyperparameters based on recommendation performance on a validation set.

3.5 Optimization

We use the stochastic gradient descent (SGD) algorithm to optimize objective (11). A previously reported sophisticated SGD strategy algorithm, which is called $LEARN_{BPR}$ [23], only considers one relation. However, our model contains item-to-context-item, item-to-context-co-cluster and co-cluster-to-context-item relations. Thus, to maximize the (11), it is intuitive to merge all kinds of relations in two domains together. In fact, the weights of different types of relations cannot be comparable to one another. Therefore, it is more reasonable to alternatively sample [16, 27] from the three training sets $\mathcal{D}_x = \{\mathcal{D}_x^s, \mathcal{D}_x^t\}$ for item-to-context-item relations, $\mathcal{D}_y = \{\mathcal{D}_y^s, \mathcal{D}_y^t\}$ for item-to-context-co-cluster relations and $\mathcal{D}_z = \{\mathcal{D}_z^s, \mathcal{D}_z^t\}$ for co-cluster-to-context-item relations. Specifically, the objective function (11) can be divided into $\mathcal{L}_1$, $\mathcal{L}_2$ and $\mathcal{L}_3$, where $\mathcal{L}_1 = \mathcal{L}(\hat{x}_{ijk}^s) + \mathcal{L}(\hat{x}_{ijk}^t)$, $\mathcal{L}_2 = \mathcal{L}(\hat{y}_{ijk}^s) + \mathcal{L}(\hat{y}_{ijk}^t)$ and $\mathcal{L}_3 = \mathcal{L}(\hat{z}_{ijk}^s) + \mathcal{L}(\hat{z}_{ijk}^t)$ are the objective function for $\mathcal{D}_x$, $\mathcal{D}_y$ and $\mathcal{D}_z$, respectively. The derivative of the loss function presented in Eq. (11) can now be expressed as:

$$\frac{\partial \mathcal{L}_1}{\partial \theta} = \frac{-e^{-\hat{x}_{ijk}^s}}{1 + e^{-\hat{x}_{ijk}^s}} \cdot \frac{\partial \hat{x}_{ijk}^s}{\partial \theta} + \frac{-e^{-\hat{x}_{ijk}^t}}{1 + e^{-\hat{x}_{ijk}^t}} \cdot \frac{\partial \hat{x}_{ijk}^t}{\partial \theta} - 2\lambda_{\theta} \cdot \theta, \tag{12}$$

$$\frac{\partial \mathcal{L}_2}{\partial \theta} = \lambda_c \Big(\frac{-e^{-\hat{y}_{ijk}^s}}{1 + e^{-\hat{y}_{ijk}^s}} \cdot \frac{\partial \hat{y}_{ijk}^s}{\partial \theta} + \frac{-e^{-\hat{y}_{ijk}^t}}{1 + e^{-\hat{y}_{ijk}^t}} \cdot \frac{\partial \hat{y}_{ijk}^t}{\partial \theta} \Big) - 2\lambda_{\theta} \cdot \theta, \tag{13}$$

$$\frac{\partial \mathcal{L}_3}{\partial \theta} = \lambda_c \Big(\frac{-e^{-\hat{z}_{ijk}^s}}{1 + e^{-\hat{z}_{ijk}^s}} \cdot \frac{\partial \hat{z}_{ijk}^s}{\partial \theta} + \frac{-e^{-\hat{z}_{ijk}^t}}{1 + e^{-\hat{z}_{ijk}^t}} \cdot \frac{\partial \hat{z}_{ijk}^t}{\partial \theta} \Big) - 2\lambda_{\theta} \cdot \theta, \tag{14}$$

where the derivatives of x_{ijk}^s, y_{ijk}^s and z_{ijk}^s for source domain s are,

$$\frac{\partial \hat{x}_{ijk}^s}{\partial \theta} = \begin{cases} q_{jf}^s - q_{kf}^s & \text{if } \theta = p_{if}^s \\ p_{if}^s & \text{if } \theta = q_{jf}^s \\ -p_{if}^s & \text{if } \theta = q_{kf}^s \\ 0 & \text{else,} \end{cases} \tag{15}$$

$$\frac{\partial \hat{y}_{ijk}^s}{\partial \theta} = \begin{cases} o_{jf}^s - o_{kf}^s & \text{if } \theta = p_{if}^s \\ p_{if}^s & \text{if } \theta = o_{jf} \\ -p_{if}^s & \text{if } \theta = o_{kf} \\ 0 & \text{else,} \end{cases} \tag{16}$$

$$\frac{\partial \hat{z}_{ijk}^s}{\partial \theta} = \begin{cases} q_{jf}^s - q_{kf}^s & \text{if } \theta = r_{if} \\ r_{if} & \text{if } \theta = q_{jf}^s \\ -r_{if} & \text{if } \theta = q_{kf}^s \\ 0 & \text{else,} \end{cases} \tag{17}$$

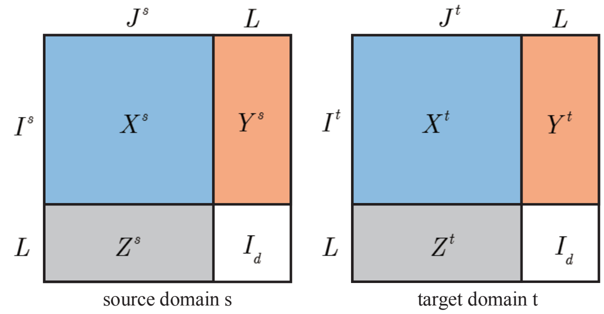


Figure 2: The CDIE-C is modelled as matrix factorization of items and co-clusters from two domains extended matrices.

where f denotes the f^{th} embedding feature of the corresponding representation vector. With reference to (15-17), the derivatives for target domain t can be obtained by swapping the superscripts s with t. They are not listed due to the page limit.

The objective function (11) can be optimized in different ways, depending on how the cross-domain correlation information, i.e., the item-to-context-co-cluster and co-cluster-to-context-item relations, is used. One solution is to train the model with sequential data from single domains (the item-to-context-item relations) and cross-domain correlative data (the item-to-context-co-cluster and co-cluster-to-context-item relations) simultaneously, which is called joint training. We summarize the detailed training algorithm in Alg. 1. An alternative solution is to learn representations with sequential data from single domains first, and then further tune representations with cross-domain correlative data, which is called two-stage training. This is inspired by the idea of pre-training and fine-tuning [2, 26]. Similarly, we summarize the training process of two-stage training in Alg. 2.

Algorithm 1 Joint Training for CDIE-C.

Initialize $\boldsymbol{\theta}$
repeat
  Draw (p_i^s, q_j^s, q_k^s) or (p_i^t, q_j^t, q_k^t) from $\{\mathcal{D}_x^s, \mathcal{D}_x^t\}$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \mu(\frac{e^{-\hat{x}^s_{ijk}}}{1+e^{-\hat{x}^s_{ijk}}} \cdot \frac{\partial \hat{x}^s_{ijk}}{\partial \boldsymbol{\theta}} + \frac{e^{-\hat{x}^t_{ijk}}}{1+e^{-\hat{x}^t_{ijk}}} \cdot \frac{\partial \hat{x}^t_{ijk}}{\partial \boldsymbol{\theta}} + 2\lambda_{\boldsymbol{\theta}} \cdot \boldsymbol{\theta})$
  Draw (p_i^s, o_j, o_k) or (p_i^t, o_j, o_k) from $\{\mathcal{D}_y^s, \mathcal{D}_y^t\}$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \mu(\frac{e^{-\hat{y}^s_{ijk}}}{1+e^{-\hat{y}^s_{ijk}}} \cdot \frac{\partial \hat{y}^s_{ijk}}{\partial \boldsymbol{\theta}} + \frac{e^{-\hat{y}^t_{ijk}}}{1+e^{-\hat{y}^t_{ijk}}} \cdot \frac{\partial \hat{y}^t_{ijk}}{\partial \boldsymbol{\theta}} + 2\lambda_{\boldsymbol{\theta}} \cdot \boldsymbol{\theta})$
  Draw (r_i, q_j^s, q_k^s) or (r_i, q_j^t, q_k^t) from $\{\mathcal{D}_z^s, \mathcal{D}_z^t\}$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \mu(\frac{e^{-\hat{z}^s_{ijk}}}{1+e^{-\hat{z}^s_{ijk}}} \cdot \frac{\partial \hat{z}^s_{ijk}}{\partial \boldsymbol{\theta}} + \frac{e^{-\hat{z}^t_{ijk}}}{1+e^{-\hat{z}^t_{ijk}}} \cdot \frac{\partial \hat{z}^t_{ijk}}{\partial \boldsymbol{\theta}} + 2\lambda_{\boldsymbol{\theta}} \cdot \boldsymbol{\theta})$
until *convergence or max-iteration has been reached*
return $\boldsymbol{\theta}$

Algorithm 2 Two-stage Training for CDIE-C.

Initialize $\boldsymbol{\theta}$
repeat
  Draw (p_i^s, q_j^s, q_k^s) or (p_i^t, q_j^t, q_k^t) from $\{\mathcal{D}_x^s, \mathcal{D}_x^t\}$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \mu(\frac{e^{-\hat{x}^s_{ijk}}}{1+e^{-\hat{x}^s_{ijk}}} \cdot \frac{\partial \hat{x}^s_{ijk}}{\partial \boldsymbol{\theta}} + \frac{e^{-\hat{x}^t_{ijk}}}{1+e^{-\hat{x}^t_{ijk}}} \cdot \frac{\partial \hat{x}^t_{ijk}}{\partial \boldsymbol{\theta}} + 2\lambda_{\boldsymbol{\theta}} \cdot \boldsymbol{\theta})$
until *convergence or max-iteration has been reached*
repeat
  Draw (p_i^s, o_j, o_k) or (p_i^t, o_j, o_k) from $\{\mathcal{D}_y^s, \mathcal{D}_y^t\}$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \mu(\frac{e^{-\hat{y}^s_{ijk}}}{1+e^{-\hat{y}^s_{ijk}}} \cdot \frac{\partial \hat{y}^s_{ijk}}{\partial \boldsymbol{\theta}} + \frac{e^{-\hat{y}^t_{ijk}}}{1+e^{-\hat{y}^t_{ijk}}} \cdot \frac{\partial \hat{y}^t_{ijk}}{\partial \boldsymbol{\theta}} + 2\lambda_{\boldsymbol{\theta}} \cdot \boldsymbol{\theta})$
  Draw (r_i, q_j^s, q_k^s) or (r_i, q_j^t, q_k^t) from $\{\mathcal{D}_z^s, \mathcal{D}_z^t\}$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \mu(\frac{e^{-\hat{z}^s_{ijk}}}{1+e^{-\hat{z}^s_{ijk}}} \cdot \frac{\partial \hat{z}^s_{ijk}}{\partial \boldsymbol{\theta}} + \frac{e^{-\hat{z}^t_{ijk}}}{1+e^{-\hat{z}^t_{ijk}}} \cdot \frac{\partial \hat{z}^t_{ijk}}{\partial \boldsymbol{\theta}} + 2\lambda_{\boldsymbol{\theta}} \cdot \boldsymbol{\theta})$
until *convergence or max-iteration has been reached*
return $\boldsymbol{\theta}$

Table 1: Statistics of cross-domain datasets for evaluation.

Datasets	Movie-Music		Health-Beauty		Clothing-Sports	
Domains	Movie	Music	Health	Beauty	Clothing	Sports
Events	626,537	200,923	165,502	101,909	173,277	186,119
Items	10,685	6,580	4,838	3,560	8,164	5,805
Sessions	232,727	78,700	71,063	42,803	77,510	76,419

4 EXPERIMENTS

In this section, we conduct extensive experiments on three cross-domain datasets to answer the following three research questions:

RQ1 How does our proposed CDIE-C perform as compared with other state-of-the-art competitors?
RQ2 How does our proposed CDIE-C method perform in handling the sparsity problem?
RQ3 How is the recommendation performance of CDIE-C under different cross-domain correlation thresholds?

4.1 Experimental Settings

4.1.1 Datasets Construction. We evaluate our method on the publically available Amazon[3] dataset [21]. We build three pairs of domains "Movies & TV - CDs & Vinyl" (Movie-Music), "Health & Personal Care - Beauty" (Health-Beauty) and "Clothing, Shoes & Jewelry - Sports & Outdoors" (Clothing-Sports) for experiments. We manually partition the interaction data of users' histories into single-domain sessions by using an one-hour idle threshold in all domains. Then, we hide user information of user-item interactions. In order to better validate the performance of our method, we keep the ratings of three or higher and select items which have more than 20 ratings. Besides, we remove sessions having less than 2 interactions. The statistics of datasets are shown in Table 1.

In our experiments, we use two different strategies to generate the training, validation, and testing sets for different evaluation settings. 1) In the overall performance comparisons for CDIE-C, test sets are built with sessions of last two months of domains, and then the remaining sessions form the training sets. We also filter out items from test sets where the items are not in training sets. Sessions whose length are less than 2 are also removed from test sets. In the meantime, we build validation sets with sessions of the last one month of training sets. Besides, for building cross-domain sessions, we link different single-domain sessions of the same users using a one-hour idle threshold in every pair of training sets. 2) In the sparsity problem handling, we conduct variants of training sets at different sparsity level ($\ell_n = [0.1, 0.3, 0.5]$) by randomly removing different percentages of records from the original training sets. Test sets and validation sets settings are consistent with 1) and the construction of cross-domain sessions is similar to 1).

4.1.2 Evaluation Methods and Metrics. After learning item representations across domains by CDIE-C, we input representations of anonymous user preferences (averages of representations of context items) one by one, and then calculate ranks of recommendable items in sessions based on cosine similarity. The evaluation metrics in our experiments are Recall@30 and Mean Reciprocal Rank (MRR)@30. While Recall@30 considers all items ranked within the first 30 to be equivalent, MRR@30 use multiplicative inverse

[3] https://www.amazon.com/

of the rank of the first correct answer among the top-30 items to emphasize the importance of a higher rank versus a lower one.

4.1.3 Comparative methods. To justify the effectiveness of our proposed CDIE-C method, we compare with several state-of-the-art single-domain recommendation methods and cross-domain recommendation methods:

Single-domain recommendation methods

(1) POP: a popularity predictor that always recommends the most popular items of training sets. Despite its simplicity, it is often a strong baseline in a single domain.
(2) ITEM2VEC [1]: an item embedding method based on the skip-gram with negative-sampling word2vec model [22]. It treats a set of items as a sequence of words.
(3) BPR [23]: one of the commonly used matrix factorization methods. It optimizes a pairwise ranking objective function via SGD.
(4) GRU4REC[4] [12]: an RNNs-based method, which introduces several modifications to classic RNNs such as a ranking loss function that makes it more viable for session-based recommendations.

Cross-domain recommendation methods

(1) CLFM [9]: a cluster-level based collaborative filtering method without user/item overlaps between domains. It can jointly learn the common rating patterns and domain-specific rating patterns.
(2) CDIE-C: our proposed method, which learns complete representations of items by jointly leveraging single-domain sessions and cross-domain sessions in a unified way. There are different variants of CDIE-C that use different combinations of item-to-context-item, item-to-context-co-cluster and co-cluster-to-context-item relations. We denote **CDIE-C (item)** for the version that focuses on the relationships of item and their contexts (i.e., context-items/context-co-clusters) ignoring co-cluster-to-context-item relations; **CDIE-C (two-stage)** learns item representations with the sequence information from single-domain sessions, and then tunes the item representations with the cross-domain correlation information (item-to-context-co-cluster and co-cluster-to-context-item relations); **CDIE-C (joint)** jointly trains the model composed of all three types of relations.

For the single-domain recommendation methods, we recommend items for the source and target domains, respectively. For cross-domain recommendation methods, we train two domains together. Besides, BPR and CLFM cannot be applied directly to session-based recommendations, due to the design for user-item interactions. However, we can overcome the problem by jointly factorizing two item-item co-occurrence matrices for two domains, and then use the average of representations of the items that occur in the session as the user preference representations.

In all of our simulations, the dimension of the latent space d is set to 100 and the number of iterations is set at 20. We monitor the convergence of the method using recall@30 on the validation set for BPR, CLFM and CDIE-C. The number of co-clusters L is set to 60. All hyperparameters and learning rates of aforementioned methods are empirically tuned based on the validation set by grid search. We repeated every experiment for 5 times to report the average results.

4.2 Overall Performance Analysis (RQ1)

To demonstrate the overall effectiveness of our proposed CDIE-C, we compare the CDIE-C with state-of-the-art single-domain and cross-domain recommendation methods.

Experimental results are shown in Table 2. We first compare the performance of single-domain recommendation methods, which use either popularity (POP), item-item co-occurrences (BPR), or the sequence information (ITEM2VEC and GRU4REC). We can see that the performance of BPR is inferior to that of ITEM2VEC and GRU4REC, as BPR is only taking into account the co-occurrences, while ignoring the order information. Besides, the GRU4REC using the sequential signals performs the best among all the single-domain recommendation methods. Nevertheless, our proposed CDIE-C outperforms single-domain competitors significantly. Taking the Movie-Music dataset as an example, CDIE-C (joint) gains improvements over GRU4REC at 12.77% (Recall@30) and 7.17% (MRR@30) for the movie domain and 10.49% (Recall@30) and 6.91% (MRR@30) for the music domain. This is because the CDIE-C method can learn complete item features by fusing the correlative data of the source and target domains and enhancing information exchange across domains with three types of relations (i.e., item-to-context-item, item-to-context-co-cluster and co-cluster-to-context-item relations), while the GRU4REC can only utilize single-domain data with item-item relations.

Next, we compare the performance of cross-domain recommendation methods, including CLFM and different variants of CDIE-C. CDIE-C (joint) consistently outperforms CDIE-C (item), verifying that incorporating co-cluster-to-context-item relations can extend the impact of cross-domain correlations and improve the quality of the recommendations. CDIE-C (joint) also outperforms CDIE-C (two-stage). This shows that jointly training with single-domain and cross-domain data is much more effective compared to separating them into two phases. Besides, the performance of CLFM is inferior to that of all variants of CDIE-C, due to ignoring the order between items.

4.3 Performance Analysis w.r.t. Sparsity (RQ2)

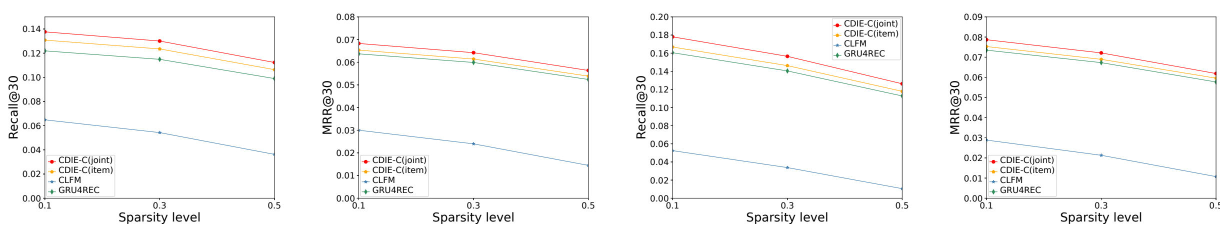
Because all competing methods are based on history records of items, they often suffer from the sparsity problem. Therefore, we explore the performance for different sparsity levels. Results[5] are displayed in Figures 3, 4 and 5. Experimental results demonstrate the effectiveness of CDIE-C in handling the sparsity problem. Take the Movie-Music dataset as an example, CDIE-C (joint) achieves the Recall@30 of 0.1123 and 0.1262 in two domains, respectively, which gains improvements over GRU4REC at 13.43% and 11.88%,respectively, when the sparsity level is equal to 0.5. By taking a closer look at the sparse training sets, we notice that there are 63.32% and 64.66% sessions with one item in the Movie domain and the Music domain, respectively. These short sessions degenerate the performance of methods (GRU4REC and CLFM). CDIE-C can extract information

[4] https://github.com/hidasib/GRU4Rec

[5] GRU4REC cannot be directly applied in the case of the sparsity problem, as new items come along when records are removed. Then, we extend it through a simple and effective way—the random guess. Therefore, we combined results of the random guess with those predicted by GRU4REC.

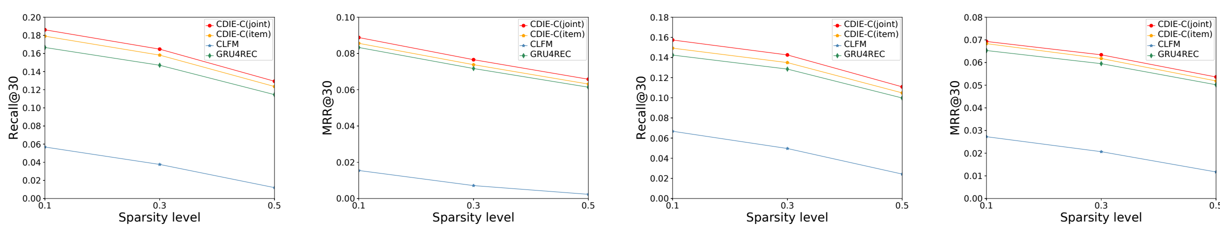
Table 2: Overall Performance Analysis on Three Cross-domain Datasets.

Methods	Movie-Music				Health-Beauty				Clothing-Sports			
	Movie		Music		Health		Beauty		Clothing		Sports	
	Recall@30	MRR@30	Recall@30	MRR@30	Recall@30	MRR@30	Recall@30	MRR@30	Recall@30	MRR@30	Recall@30	MRR@30
POP	0.0426	0.0065	0.0272	0.0051	0.0598	0.0098	0.0728	0.0129	0.0325	0.0045	0.0477	0.0098
BPR	0.0664	0.0231	0.079	0.0276	0.0778	0.0269	0.0658	0.0245	0.0231	0.0084	0.0435	0.0154
ITEM2VEC	0.0763	0.0336	0.0921	0.0432	0.0894	0.0418	0.0867	0.0369	0.0344	0.0202	0.0559	0.0287
GRU4REC	0.1261	0.0642	0.1745	0.0825	0.1677	0.0869	0.1609	0.0711	0.0964	0.0346	0.1241	0.0505
CLFM	0.0756	0.0331	0.0874	0.0365	0.0845	0.0375	0.0815	0.0352	0.0303	0.0198	0.0498	0.0231
CDIE-C(item)	0.1353	0.0652	0.1813	0.0841	0.1801	0.0893	0.1683	0.0747	0.0983	0.0349	0.1316	0.0512
CDIE-C(two-stage)	0.1372	0.0657	0.1856	0.0852	0.1825	0.0896	0.1691	0.0751	0.0994	0.0351	0.1324	0.0514
CDIE-C(joint)	**0.1422**	**0.0688**	**0.1928**	**0.0882**	**0.1864**	**0.0924**	**0.1775**	**0.0755**	**0.1029**	**0.0357**	**0.1373**	**0.0532**



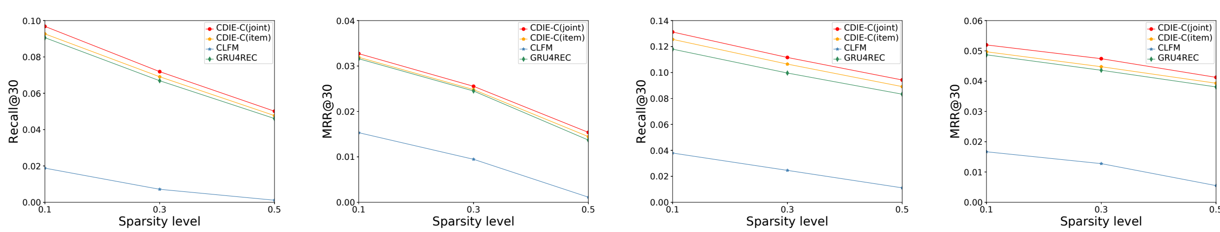
(a) Movie (b) Movie (c) Music (d) Music

Figure 3: Performance w.r.t sparsity on the Movie-Music dataset.



(a) Health (b) Health (c) Beauty (d) Beauty

Figure 4: Performance w.r.t sparsity on the Health-Beauty dataset.



(a) Clothing (b) Clothing (c) Sports (d) Sports

Figure 5: Performance w.r.t sparsity on the Clothing-Sports dataset.

in short sessions by linking them with sessions in other domains, while GRU4REC and CLFM cannot utilize these sessions with only one item. Therefore, the missing information of items can be supplemented via related domains. Besides, the performance of CDIE-C (joint) is superior to that of CDIE-C (item). CDIE-C (item) does not take advantage of co-cluster-to-context-item relations. These relations are very useful to expand influence of cluster-level correlations, as they can link more items by the co-cluster-to-context-item relations. The performance of CLFM is still inferior to that of GRU4REC. It is because CLFM cannot learn effective cluster-level

Table 3: Performance analysis w.r.t. correlation thresholds on the Movie-Music dataset.

Domains	Metrics	Methods	Correlation Thresholds			
			0.1	0.15	0.2	0.25
Movie	Recall@30	CDIE-C(item)	0.1291	0.1332	0.1353	0.1316
		CDIE-C(joint)	0.1361	0.1398	0.1422	0.1387
	MRR@30	CDIE-C(item)	0.0512	0.0634	0.0652	0.0628
		CDIE-C(joint)	0.0456	0.0475	0.0688	0.0637
Music	Recall@30	CDIE-C(item)	0.1736	0.1782	0.1813	0.1761
		CDIE-C(joint)	0.1943	0.1989	0.1928	0.1876
	MRR@30	CDIE-C(item)	0.0792	0.0814	0.0841	0.0796
		CDIE-C(joint)	0.0824	0.0858	0.0882	0.0847

patterns across domains, as its performance depends on abundant item-item co-occurrences.

4.4 Performance Analysis w.r.t. Correlation Thresholds (RQ3)

By setting a suitable threshold p, we can improve not only the computational efficiency with a reduced number of co-clusters but also the robustness by filtering out noise. It is critical to find the appropriate co-clusters that can link two different domains for fusing the effective complementary information in cross-domain recommendations. Therefore, we need to conduct experiments to study the impact of p, which can help filter out noise of cross-domain information. Table 3 presents the performance of our proposed CDIE-C (item) and CDIE-C (joint) in terms of recall@30 and mrr@30 with different threshold p on the Movie-Music dataset. Due to space limitation, we omit the results on other datasets, which are similar. From the results, we observe that the performance is sensitive to p. First, the performance of the two methods improves with the increase of p and achieves the best performance when $p = 0.2$. This is because useful co-clusters can be found with a lower threshold p. Then, the increase of p leads to the decrease of performance, as noise is also introduced with a higher threshold. To achieve the best performance, we set $p = 0.2$.

5 CONCLUSION

In this paper, we aim to learn cross-domain comprehensive representations of items for the sparsity problem in session-based recommendations, and propose a cross-domain item embedding method based on co-clustering to address these challenging issues, including noise and diversity of information. We conducted extensive experiments to evaluate the performance of CDIE-C on three cross-domain datasets. The results showed CDIE-C is significantly better than the state-of-the-art methods (up to 13.43% enhancement for Recall@30 comparing to GRU4REC), when the sparsity level is equal to 0.5 on datasets. Therefore, our method can work well even with a sparse system, where historical data are lacking or are insufficient for items in sessions. The performance of CDIE-C can be reinforced by fully exploring the correlations between different domains, which can provide complementary information to one another. For future works, we may consider integrating more types of information into CDIE-C for improving the performance of embedding and extending our method to multiple domains.

REFERENCES

[1] Oren Barkan and Noam Koenigstein. 2016. Item2vec: neural item embedding for collaborative filtering. In *MLSP*. IEEE, 1–6.
[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. 2013. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence* 35, 8 (2013), 1798–1828.
[3] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. 2013. Recommender systems survey. *Knowledge-based systems* 46 (2013), 109–132.
[4] Iván Cantador, Ignacio Fernández-Tobías, Shlomo Berkovsky, and Paolo Cremonesi. 2015. Cross-domain recommender systems. In *Recommender Systems Handbook*. Springer, 919–959.
[5] Da Cao, Liqiang Nie, Xiangnan He, Xiaochi Wei, Shunzhi Zhu, and Tat-Seng Chua. 2017. Embedding Factorization Models for Jointly Recommending Items and User Generated Lists. In *SIGIR*. ACM, 585–594.
[6] Inderjit S Dhillon. 2001. Co-clustering documents and words using bipartite spectral graph partitioning. In *KDD*. ACM, 269–274.
[7] Ignacio Fernández-Tobías, Iván Cantador, Marius Kaminskas, and Francesco Ricci. 2011. A generic semantic-based framework for cross-domain recommendation. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*. ACM, 25–32.
[8] R Michael Furr and David C Funder. 2004. Situational similarity and behavioral consistency: Subjective, objective, variable-centered, and person-centered approaches. *Journal of Research in Personality* 38, 5 (2004), 421–447.
[9] Sheng Gao, Hao Luo, Da Chen, Shantao Li, Patrick Gallinari, and Jun Guo. 2013. Cross-domain recommendation via cluster-level latent factor model. In *ECML-PKDD*. Springer, 161–176.
[10] Mihajlo Grbovic, Vladan Radosavljevic, Nemanja Djuric, Narayan Bhamidipati, Jaikit Savla, Varun Bhagwan, and Doug Sharp. 2015. E-commerce in your inbox: Product recommendations at scale. In *KDD*. ACM, 1809–1818.
[11] Asnat Greenstein-Messica, Lior Rokach, and Michael Friedman. 2017. Session-based recommendations using item embedding. In *IUI*. ACM, 629–633.
[12] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. *ICLR* (2015).
[13] Tomoharu Iwata and Takeuchi Koh. 2015. Cross-domain recommendation without shared users or items by sharing latent vector distributions. In *Artificial Intelligence and Statistics*. 379–387.
[14] Dietmar Jannach and Malte Ludewig. 2017. When recurrent neural networks meet the neighborhood for session-based recommendation. In *RecSys*. ACM, 306–310.
[15] Marius Kaminskas, Ignacio Fernández-Tobías, Iván Cantador, and Francesco Ricci. 2013. Ontology-based identification of music for places. In *Information and Communication Technologies in Tourism 2013*. Springer, 436–447.
[16] Artus Krohn-Grimberghe, Lucas Drumond, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2012. Multi-relational matrix factorization using bayesian personalized ranking for social network data. In *WSDM*. ACM, 173–182.
[17] Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *NIPS*. 2177–2185.
[18] Bin Li, Qiang Yang, and Xiangyang Xue. 2009. Can Movies and Books Collaborate? Cross-Domain Collaborative Filtering for Sparsity Reduction.. In *IJCAI*, Vol. 9. 2052–2057.
[19] Dawen Liang, Jaan Altosaar, Laurent Charlin, and David M Blei. 2016. Factorization meets the item embedding: Regularizing matrix factorization with item co-occurrence. In *RecSys*. ACM, 59–66.
[20] Greg Linden, Brent Smith, and Jeremy York. 2003. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing* 7, 1 (2003), 76–80.
[21] Julian McAuley, Rahul Pandey, and Jure Leskovec. 2015. Inferring networks of substitutable and complementary products. In *KDD*. ACM, 785–794.
[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).
[23] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *UAI*. AUAI Press, 452–461.
[24] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *WWW*. ACM, 285–295.
[25] Yong Kiam Tan, Xinxing Xu, and Yong Liu. 2016. Improved recurrent neural networks for session-based recommendations. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. ACM, 17–22.
[26] Jian Tang, Meng Qu, and Qiaozhu Mei. 2015. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *KDD*. ACM, 1165–1174.
[27] Yaqing Wang, Chunyan Feng, Ling Chen, Hongzhi Yin, Caili Guo, and Yunfei Chu. 2018. User identity linkage across social networks via linked heterogeneous network embedding. *World Wide Web* (2018), 1–22.
[28] Fuzhen Zhuang, Yingmin Zhou, Fuzheng Zhang, Xiang Ao, Xing Xie, and Qing He. 2017. Sequential Transfer Learning: Cross-domain Novelty Seeking Trait Mining for Recommendation. In *WWW*. 881–882.