# Knowledge-aware Neural Collective Matrix Factorization for Cross-domain Recommendation

Li Zhang[1], Yan Ge [2], Jun Ma[3], Jianmo Ni[4] and Haiping Lu[1]

[1]Department of Computer Science, University of Sheffield, Sheffield, United Kingdom
[2] Department of Computer Science, University of Bristol, Bristol, United Kingdom
[3] Amazon Inc., Seattle, WA, USA
[4] Google Inc., USA

[1]{lzhang72, h.lu}@sheffield.ac.uk, [2]yan.ge@bristol.ac.uk, [3]junmaa@amazon.com, [4] jianmon.@google.com

## ABSTRACT

Cross-domain recommendation (CDR) can help customers find more satisfying items in different domains. Existing CDR models mainly use common users or mapping functions as bridges between domains but have very limited exploration in fully utilizing extra knowledge across domains. In this paper, we propose to incorporate the knowledge graph (KG) for CDR, which enables items in different domains to share knowledge. To this end, we first construct a new dataset *AmazonKG4CDR* from the Freebase KG and a subset (two domain pairs: movies-music, movie-book) of Amazon Review Data. This new dataset facilitates linking knowledge to bridge within- and cross-domain items for CDR. Then we propose a new framework, KG-aware Neural Collective Matrix Factorization (KG-NeuCMF), leveraging KG to enrich item representations. It first learns item embeddings by graph convolutional autoencoder to capture both domain-specific and domain-general knowledge from adjacent and higher-order neighbours in the KG. Then, we maximize the mutual information between item embeddings learned from the KG and user-item matrix to establish cross-domain relationships for better CDR. Finally, we conduct extensive experiments on the newly constructed dataset and demonstrate that our model significantly outperforms the best-performing baselines.

## KEYWORDS

Cross-domain recommendation, knowledge graph, graph autoencoder.

## 1 INTRODUCTION

Cross-domain recommendation (CDR) [8] is a promising solution to the data sparsity problem in recommender systems. Conventional single-target CDR models leverage information from a richer (source) domain to improve the recommendation performance in a sparser (target) domain [2, 13, 41]. To improve performance in both domains, recent dual-target CDR models [20, 23, 45] are proposed, which enables bidirectional transfer across domains with dual-learning mechanism [10, 43].

Despite encouraging results from existing CDR models, several key issues remain unsolved [46]. Firstly, current models, including the dual-target ones, can not simultaneously improve the performance in both source and target domains due to negative transfer [25]. In general, the knowledge learned from the sparser domain is less accurate than that learned from the richer domain. Thus, the recommendation performance in the richer domain tends to decline if the transfer direction is simply inverted. Secondly, current CDR models mainly use common users [23, 45] or mapping functions [20] to build connections between domains. In real-life scenarios, relationships between items within or across domains can characterize item-wise semantic relatedness to help understand user-item interaction patterns [35]. However, current CDR models are inadequate in capturing such useful item-item relationships.

In this paper, we aim to address this gap by leveraging knowledge graph (KG), a natural bridge for items from different domains [36]. KGs can benefit the CDR task in multiple ways [33]. First, rich and explicit connections among items in the KG can help improve the recommendation performance in each domain, particularly the sparser domain. As shown in Fig 1, a user who has watched "*Harry Potter and the Deathly Hallows*" is very likely to have interest in the movie "*Fantastic Beasts and Where to Find Them*" (directed by the same director), which can be recommended with the assistance of domain-specific knowledge in the KG. Second, domains often share some domain-general information. For example, genre can characterize both book and movie domains. "*Lord of the Ring*" (from movies), "*Harry Potter*" (from books) can be closely connected in the KG via the related genre *Fantasy*. KGs provide a natural bridge to build connections between domains. Leveraging such information can help models understand target or source items by associating rich semantic relatedness among items from different domains and further improve recommendation performance.

To build KG-aware CDR, three unique technical challenges arise. (1) Though several datasets exist for KG-aware single-domain recommendation, no publicly-available dataset exists for KG-aware
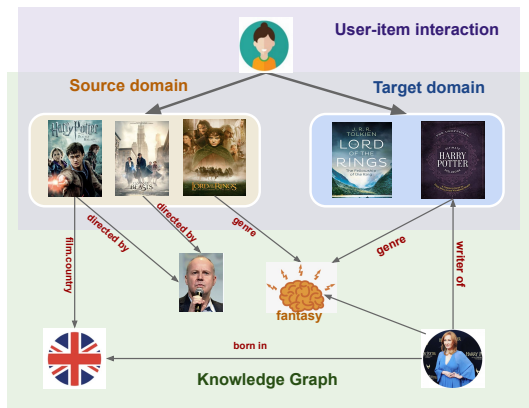
**Figure 1: Knowledge graph is a natural bridge that connects items from different domains. For example, "*Lord of the Ring*" in movies can get connected with "*Harry Potter*" in books via related genre *Fantasy*. Such inter-domain knowledge can reveal similar semantic relations among items from different domains to further improve cross-domain recommendation. This paper constructs a new dataset and proposes a new model to achieve this goal.**

CDR. (2) To improve CDR, item (entity) embeddings (representations) learned from the KG should contain both domain-specific and domain-general information, which typically comes from different hops of neighbors in KG. The second challenge is to model both adjacent and higher-order relations in the item representation learning process. (3) Item embeddings learned from the KG and those from the user-item interaction matrix should be closely related, e.g., highly correlated, so that cross-domain relationships can be effectively established. How to ensure this is the third challenge we need to overcome.

To address the challenges above, we construct a new dataset for KG-aware CDR and propose a novel KG-aware Neural Collective Matrix Factorization (KG-aware NeuCMF) model. Firstly, we construct a new dataset named *Amazon product Knowledge Graph for CDR* (*AmazonKG4CDR*) using a subset (movie, book, and music) of the Amazon Review Data (2018) [24] and the Freebase KG [3, 5]. Then, we propose a two-step framework for KG-aware CDR. 1) We train a shared autoencoder using a relational graph convolutional network (RGCN) on the knowledge graph following a contrastive learning-style [16, 29]. GCN-based encoders learn a node's embedding by aggregating information from its neighbors via non-linear transformation and aggregation [17]. Long-range node dependencies can be captured by stacking multiple GCN layers to propagate information for multiple hops [39]. This enables capturing both domain-specific and domain-general information from different hops of neighbors in the KG. 2) To establish cross-domain relationships, the embeddings learned from KG should be highly coherent with those from the user-item interaction matrix. Therefore, we incorporate the mutual information (MI) estimation [1] into the neural collective matrix factorization (NeuCMF) framework. This mechanism allows our model to preserve both user-item interaction and KG information across items. Finally, we conduct extensive experiments on our newly constructed datasets and demonstrate that

our model significantly outperforms the best-performing baselines, with up to 21% (movie), 15.18% (music) improvement, in terms of the mean absolute error (MAE) in movie-music domains recommendation.

In summary, our contributions are threefold:

- We construct and leverage the knowledge graph for CDR task. To the best of our knowledge, this is the first time to apply KG information for CDR.
- We propose a two-step KG-aware NeuCMF framework for KG-aware CDR, which enables learned item embeddings can capture both user-item interactions, domain-general, domain-specific information from the KG.
- We conduct extensive experiments on the newly constructed datasets. Experimental results show that our proposed model can significantly outperform most state-of-the-art CDR models.

## 2 RELATED WORK

### 2.1 Cross-Domain Recommendation

Different from conventional single-domain recommendation, CDR can leverage information from source domain to improve the performance of target domain [2, 8], namely single-target CDR, which is a powerful tool to deal with the data sparsity problem. These approaches extend the single-domain recommendation models by utilizing same contents, such as tags, reviews [7, 42], common items or users [12, 21, 30] as the bridge between and transfer information between domains [13, 22, 27, 28].

The single-target CDR approaches only focus on how to leverage the source domain to help improve the recommendation accuracy on the target one, but not vice versa. Recently, dual-target CDR mothods [20, 23, 45] has been proposed to improve the performance on both source and target domains simultaneously by leveraging dual-transfer learning strategies [10, 43]. However, as referred to as Negative Transfer [25], this idea does not work, because the knowledge learned from the sparser domain is less accurate than that learned from the richer domain, thus the recommendation accuracy on the richer domain is more likely to decline by simply and directly changing the transfer direction. Therefore, dual target CDR demands novel and effective solutions. None of the current CDR models can indeed improve the performance on both domains simultaneously, and they are significantly hindered by limited information and connections between two domains.

### 2.2 Knowledge Graph for Recommendation

In recent years, introducing recommendations with the KG as side information has attracted considerable interest [33, 36, 37]. A KG is a heterogeneous graph, where nodes represent as entities, edges represent relations between entities and a fact in KG is usually represented in the form of a triple (*head entity*, *relation*, *tail entity*) [36]. KGs contain rich semantic relatedness among items and incorporating KGs in RS can help explore the latent connections and provide explanations for recommended items [9]. Currently, KG-aware RS models are only for the single-domain RS [4, 31, 33, 36, 37, 44]. While one bottleneck for CDR is lacking of connections between domains. since KGs can naturally connect different domains, it would be
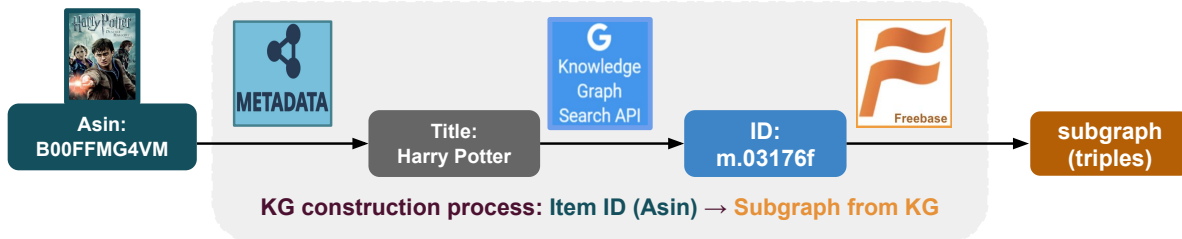
Figure 2: KG construction for Amazon products.

promising by incorporating KG in the user-item interaction matrix for better cross-domain recommendation performance.

## 3 KG-AWARE NEUCMF MODELS

In this section, we present the technical details of our proposed CDR model, KG-aware Neural CMF (KG-NeuCMF) that aims to improve the performance of CDR by leveraging the KG. This section first introduces how to construct the knowledge graph for items. Then we formulate the task and present our proposed framework: KG-NeuCMF.

### 3.1 KG Construction for CDR

To develop a knowledge-aware CDR system, a key issue is how to obtain rich and structured knowledge information for items. Existing research works use side information from the original recommender system, such as tags and reviews. We argue that the KG information will provide additional useful information to the CDR task, since the intra-domain relationship among items can be captured. In this paper, we present *AmazonKG4CDR V1.0*, a new dataset linking KG information for CDR, which can be useful for researchers in the related areas to explore possible approaches with the rich KG information.

We use the widely used dataset, Amazon Review Data (2018) [24], covering various domains, from which we select a subset that includes two domain pairs: movie-music, movie-book, which are being linked together through a common user ID identifying the same user. On the KG side, we use the well-known KG: Freebase [3]. It stores facts by triples of the form $< head >< relation ><$ $tail >$. Since Freebase shut down its services, we use its latest public version. We map items into Freebase entities via title matching if there is a mapping available. Fig.2 shows the whole linkage process. Since we only have item Asins (IDs of Amazon products), we need to get items' titles from the Amazon Review metadata first[1]. These titles are later used to get KG entity IDs from *The Knowledge Graph Search API*, which are used to extract the graph information from Freebase.

During the linkage process, we have dealt with several problems that will affect the quality of the extract knowledge graph. First, the correctness of the extracted KG entity IDs should be ensured. For example, a query is *"Harry Potter"* (a book name), and returned results can be both movies and books. So, we filter returned results

by their type and name to ensure extracted IDs are correct. To ensure the KG quality, we preprocess the extracted KG by filtering out infrequent entities (e.g., lower than 10 in both datasets) and retaining the relations appearing in at least 100 triplets.

### 3.2 Problem Statement

In this paper, we study the problem of KG-aware CDR. Formally, we are given two domains, a source domain $\mathcal{S}$ (e.g., movie recommendation) and a target domain $\mathcal{T}$ (e.g., book recommendation) that can be represented as two user-item interaction matrices $\mathbf{R}_{\mathcal{S}}$ and $\mathbf{R}_{\mathcal{T}}$, where $r_{ui} = 1$ indicates that user $u$ engages with item $i$, otherwise $r_{ui} = 0$. In real online shopping platforms (e.g., Amazon), users in domain $\mathcal{S}$ and domain $\mathcal{T}$ often overlap, meaning that they have purchased items in both domains. The set of users in both domains are shared, denoted by $\mathcal{U}$ (of size $m = |\mathcal{U}|$). In our setting, there is no overlap of items between two domains and each item only belongs to one single domain. Denote the set of items in $\mathcal{S}$ and $\mathcal{T}$ by $\mathcal{I}_{\mathcal{S}}$ and $\mathcal{I}_{\mathcal{T}}$ with size $n_{\mathcal{S}} = |\mathcal{I}_{\mathcal{S}}|$) and $n_{\mathcal{T}} = |\mathcal{I}_{\mathcal{T}}|$ respectively. Additionally, we also have a knowledge graph $\mathcal{G}$, a multi-relational graph, containing rich facts about items. Each fact in the KG is represented as a triple (head entity,relation,tail entity) $((h, r, t))$ [36]. The KG can represent large-scale information from multiple domains [6]. In recommendation scenarios, an item in the user-item interaction matrix corresponds to an entity in the KG.

Given $\mathbf{R}_{\mathcal{S}}$ and $\mathbf{R}_{\mathcal{T}}$ as well as the knowledge graph $\mathcal{G}$, we aim to predict whether user $u$ will engage with item $i$ with which the user has no interaction before. Our goal is to learn a prediction function $\hat{y}_{ui} = f(u, i \mid \Theta, \mathbf{R}_{\mathcal{S}}, \mathbf{R}_{\mathcal{T}}, \mathcal{G})$, where $\hat{y}_{ui}$ denotes the probability (or the rating score) that user $u$ will engage with item $i$ and $\Theta$ denotes the model parameters of function $f$.

### 3.3 Methodology

In this subsection, we present the technical details of our proposed model, KG-aware Neural CMF (KG-NeuCMF) that aims to improve the performance of CDR by leveraging the KG. Fig.3 shows the overview of the proposed framework. In the first stage, we propose to learn KG-level representations by exploiting a multi-layer RGCN [29] through the encode-decode paradigm by minimizing the reconstruction loss that follows a contrastive learning-style convention [18]. This step aims to learn item embeddings containing both domain-specific and domain-general information from different hops of neighbors in KG. In the second-stage, we learn item and user embeddings by borrowing ideas from the CMF framework

---

[1]https://nijianmo.github.io/amazon/index.html
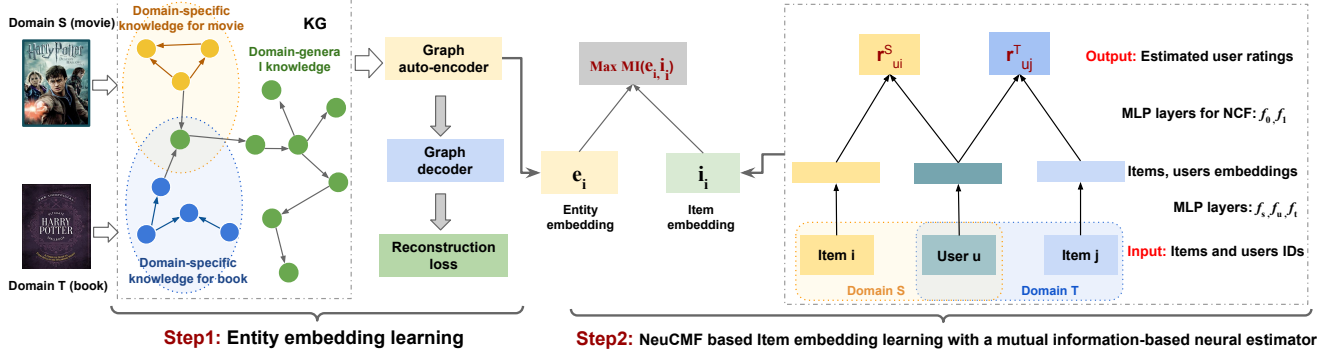
**Figure 3: The framework of our model: KG-aware NeuCMF. It learns item representations from both KG (left) and user-item interaction matrices (right). Entity (item) representations learned from KG contain both domain-specific and domain-general information by utilizing graph autoencoding strategy, which can help assist the CDR task. Item embeddings are learned by a neural CMF model. To ensure the two types of embeddings are highly correlated, we maximize their MI by the neural mutual information estimator (middle).**

[30] and neural CF (NCF) [11]. Instead of jointly factorizing the two user-item interaction matrices directly as in CMF, we propose to utilize neural networks to jointly learn the two matrices by sharing user latent representations. Finally, item representations learned from KG and user-item interaction matrix should be highly correlated. To quantify such correlation, we also exploit to maximize MI [1] between the two types of representations.

*3.3.1 Entity embedding learning.* To utilize the KG in our task, we first need to learn entity representations. We do this by training a graph autoencoder model in the unsupervised fashion and learn representations in an encode-decode paradigm [18, 29]. We employ RGCN [29] as our encoder that learns an entity embedding by aggregating information from its adjacent neighbors via non-linear transformation and aggregation dependent on the connecting relation, which can be denoted as

$$f_{en}(\mathbf{e}_i^{(l)}, \mathbf{e}_j^{(l)}) = \sigma(\mathbf{W}_0^{(l)}\mathbf{e}_i^{(l)} + \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{N}_i^r} \frac{1}{c_{ij}} \mathbf{W}_r^{(l)}\mathbf{e}_j^{(l)}), \quad (1)$$

where $\mathbf{e}_i^{(l)}$, $\mathbf{e}_j^{(l)}$ are the hidden state of node $i$ and node $j$ in the $l-$th layer of the encoder, $\sigma$ is an activation function such as ReLU, $\mathbf{W}_0^{(l)}$, $\mathbf{W}_r^{(l)}$ are (learnable parameters) relation-specific transformation mapping matrices depending on the type of edge, $c_{ij}$ is problem-specific normalization constant that can either be learned or chosen in advance, and $\mathcal{N}_i^r$ denotes the set of neighbors of node $i$ under relation $r \in \mathcal{R}$. Through this operation, the local proximity structure and related semantic information can be successfully captured and stored in the new representation of each entity. Long-range node dependencies can be captured by stacking multiple graph encoder layers and this mechanism ensures that distinct domains can be connected via the information propagation.

The decoder can be any scoring function of KG embedding methods [36] that are used to measure the plausibility of each fact $(h, r, t)$. Following [29], we use DisMult [40] factorization as the scoring function, which is well known for its simplicity and efficiency and a triple $(h, r, t)$ is scored as

$$f_{de}(\mathbf{e}_h, \mathbf{r}, \mathbf{e}_t) = \mathbf{e}_h \mathbf{R}_r \mathbf{e}_t, \quad (2)$$

where $\mathbf{e}_h, \mathbf{e}_t \in \mathbb{R}^d$ are encoded features vector for entity $h$ and $t$, and each relation $\mathbf{r}$ is associated with a diagonal matrix $\mathbf{R}_r \in \mathbb{R}^{d \times d}$.

We train the encoder and decoder with negative sampling. We construct an equal number of negative samples by randomly replacing the head entity or tail entity of each positive sample and the overall set of samples are denoted by $\mathcal{M}$. Then we minimize the cross-entropy loss of positive and negative node pairs

$$\mathcal{L} = \sum_{(\mathbf{e}_h, \mathbf{r}, \mathbf{e}_t, y) \in \mathcal{M}} (y \log f_{de}(\mathbf{e}_h, \mathbf{r}, \mathbf{e}_t))) + (1-y) \log(1 - f_{de}(\mathbf{e}_h, \mathbf{r}, \mathbf{e}_t)). \quad (3)$$

*3.3.2 NeuCMF module.* Typically the user-item interaction matrices are highly sparse and it is beneficial to learn them simultaneously [30]. Collective matrix factorization (CMF) jointly factorizes two matrices by sharing the user latent factors. Motivated by neural CF (NCF) [11], we propose to utilize neural networks to jointly learn the two matrices by sharing user latent representations as shown in Fig. 3. The predicted scores in two domains are

$$r_{ui}^{\mathcal{S}} = f_0(f_u(\mathbf{u}_u), f_s(\mathbf{i}_i^{\mathcal{S}}), \quad (4)$$

$$r_{uj}^{\mathcal{T}} = f_1(f_u(\mathbf{u}_u)), f_t(\mathbf{i}_j^{\mathcal{T}})), \quad (5)$$

where $\mathbf{u}_u$, $\mathbf{i}_i^{\mathcal{S}}$ and $\mathbf{i}_j^{\mathcal{T}}$ are represented one-hot vectors of users, items from domain $\mathcal{S}$ and domain $\mathcal{T}$ respectively. Only the element corresponding to that index is 1 and all others are 0. $f_u$, $f_s$ and $f_t$ can be multi-layer perceptron (MLP) that project sparse representations to dense vectors. The obtained embeddings are then feed into two separate multi-layer neural architectures to map the latent vectors to predict scores $r_{us}^{\mathcal{S}}$, $r_{ut}^{\mathcal{T}}$ for the two domains. Given $\mathbf{R}_{\mathcal{S}}$ and $\mathbf{R}_{\mathcal{T}}$, we minimize the two reconstruction losses $\mathcal{L}_{\mathcal{S}}$ and $\mathcal{L}_{\mathcal{T}}$ with the predicted scores.

The NeuCMF module connects two domains only by the common users, and fails to capture the relations among items. The item embedding learned from KG can capture both domain-specific and

domain-general knowledge, thus will be effective for both single-domain and cross-domain recommendation. Intuitively, the learned item embedding from user-item interaction matrices should be highly correlated to the KG-level embeddings. Therefore, this motivates us to exploit to maximize MI [1] between the two types of representations to guarantee their highly correlated relationship. We design our neural mutual information estimator based on a discriminator $\mathcal{D}(x, y)$ for their pairwise relationships, to provide probability scores for sampled pairs. To be specific, we generate positive samples as $(\mathbf{e}_i, \mathbf{i}_i)$ ($\mathbf{i}$ can come from domain $\mathcal{S}$ and domain $\mathcal{T}$, half-half) and negative samples are generated by associating sampled items with fake embeddings based on shuffling strategy [32]. We define the loss function as:

$$\mathcal{L}_{mul} = -\frac{1}{N}\left(\sum_{i=1}^{N_{pos}} \mu(\mathbf{i}_i, \mathbf{e}_i) log\sigma(\mathbf{i}_i, \mathbf{e}_i) + \sum_{i=1}^{N_{neg}} \mu(\tilde{\mathbf{i}}_i, \mathbf{e}_i) log\sigma(\tilde{\mathbf{i}}_i, \mathbf{e}_i)\right),$$
(6)

where $N = N_{pos} + N_{neg}$, $N_{pos}$, $N_{neg}$ denotes the number of positive and negative samples, $\mu(\cdot)$ is an indicator function, $\sum_{i=1}^{N_{pos}} \mu(\mathbf{i}_i, \mathbf{e}_i) = 1$ and $\sum_{i=1}^{N_{neg}} \mu(\tilde{\mathbf{i}}_i, \mathbf{e}_i) = 1$ corresponds to positive and negative pair samples. We aim to minimize $\mathcal{L}_{mul}$, which is equivalent to maximize the mutual information, to jointly preserve the KG-level and user-item interaction information.

The final loss includes: the loss ($\mathcal{L}_S$) of source and loss ($\mathcal{L}_T$) of target recommendation with the mutual information maximization loss $\mathcal{L}_{mul}$. The objective is to minimize the overall loss $\mathcal{L}$ as follows:

$$\mathcal{L} = \mathcal{L}_S(\Theta_S) + \mathcal{L}_T(\Theta_T) + \mathcal{L}_{mul}(\Theta_{mul}) + \lambda\|\Theta\|,$$
(7)

where $\Theta = \Theta_S \cup \Theta_T \cup \Theta_{\mathcal{L}_{mul}}$. Note that $\Theta_S$ and $\Theta_T$ share user embeddings. The objective function can be optimized by stochastic gradient descent (SGD) and its variants like adaptive moment method (Adam) [15].

# 4 EXPERIMENT

## 4.1 Dataset

We use the Amazon Review Data (2018) [24] that is widely used for product recommendation. It contains users' rate (ranging from 1 to 5) for product from various domains. We select a subset that includes two domain pairs: movie-music(MM), movie-book(MB), which are being linked together through a common user ID identifying the same user. We construct the knowledge graph for each item by utilizing Freebase and take triplets that involve two-hop neighbor entities of items into consideration. The basic statistics details are presented in Table 1. The recommendation task can be formulate as the regression (rating) or the binary classification (recommend or not) tasks. Following [26], we evaluate the recommendation performance based MAE, F1_score (Threshold of positive rating is 4) for the regression and classification performance, respectively.

## 4.2 Baselines

To validate the performance of the proposed model, we compare the performance with five representative models, in which two single-domain RS models (MF, NCF) and three CDR models (CMF, CoNet, DDTCDR) using the publicly released implementations.

**Table 1: Statistics of the dataset.**

|              | Domain: Music-Movie | | Domain: Book-Movie | |
|--------------|--------|---------|---------|-----------|
|              | Music  | Movie   | Book    | Movie     |
| Users        | 4,196  | 4,196   | 3,977   | 3,977     |
| Items        | 7,412  | 10,919  | 11,372  | 8,118     |
| Interactions | 21,986 | 49,027  | 22,214  | 29,245    |
| Entities     | 85,612 | 387,178 | 258,999 | 990,141   |
| Relations    | 155    | 340     | 127     | 295       |
| Triples      | 288,731| 610,314 | 522,814 | 1,787,190 |

- **MF** [19]. Matrix Factorization (MF) is a classic latent factors CF approach which learns the user and item factors via matrix factorization in each domain separately.
- **NCF** [11]. Neural Collaborative Filtering (NCF) is a neural network architecture to model latent features of users and items using CF method. The NCF models are trained separately for each domain without transferring any information.
- **CMF** [30]. Collective Matrix Factorization (CMF) jointly factorizes matrices of each domains. In our scenarios, The shared user factors enable knowledge transfer between cross domains .
- **CoNet** [12]. Collaborative Cross Networks (CoNet) enables dual knowledge transfer across domains by introducing cross connections from one base network to another and vice versa.
- **DDTCDR** [20]. Deep Dual Transfer Cross Domain Recommendation (DDTCDR) learns latent orthogonal mappings across domains and provides cross domain recommendations by leveraging user preferences from all domains.

## 4.3 Implementation details

In the KG-pretrain step, we utilize a two-layer RGCN as the encoder to obtain entity embeddings. In the NeuCMF module, we apply one-layer neural networks to project the one-hot vectors of users, and items to low-dimensional embedding vectors and $f_0$ and $f_1$ are two one-layer neural networks to map the latent vectors to predict scores. Throughout the experiments, the embedding size is tuned in the range of [8,16,32] and we use the Adam optimizer [15] with learning rate 0.001, L2 regularization 0.0001. For each dataset, the ratio of training, evaluation, and test set is 6 : 2 : 2 [34]. We employ the early stopping strategy based on the validation accuracy with a window size of 10 (we will stop training if the validation loss does not decrease for 10 consecutive epochs) and train 200 epochs at most. We report results over 20 runs with random weight matrix initialization. For a fair comparison, we set the same hyperparameters of the baselines as our model.

## 4.4 Overall Performance of CDR

We have conducted experiments on two cross domain tasks, movie-music (MM) and movie-book (MB), and the corresponding results of our model and baselines are shown in Table 2 and Table 3. We can see that our proposed model can consistently obtain the best performance across movie-music and movie-book recommendations

Table 2: Comparison of recommendation performance in Movie-Music (%). The best results are in bold and the second best ones are <u>underlined</u>.

| Methods | Movie-Music (MM) | | | |
| | Movie | | Music | |
| | MAE | F1_Score | MAE | F1_Score |
|---|---|---|---|---|
| MF [19] | 20.94±2.54 | 74.97±4.50 | 23.79±1.69 | 72.57±0.75 |
| NCF [11] | 19.01±0.09 | 88.93±0.05 | 15.25±3.23 | <u>93.05±0.43</u> |
| CMF [30] | 20.23±1.97 | <u>89.09±0.36</u> | <u>11.66±2.35</u> | 92.45±0.36 |
| CoNET [12] | <u>18.22±0.36</u> | 88.68±0.70 | 13.96±0.36 | 92.05±0.48 |
| DDTCDR [45] | 20.69±0.35 | 74.84±1.74 | 15.82±0.75 | 89.05±2.13 |
| Ours | **14.23±0.97** | **90.69±0.22** | **9.89±0.35** | **94.45±0.32** |
| Improvement (%) | 21.28 % | 1.80 % | 15.18 % | 1.50% |

Table 3: Comparison of recommendation performance in Movie-Book(%). The best results are in bold and the second best ones are <u>underlined</u>.

| Methods | Movie-Book (MB) | | | |
| | Movie | | Book | |
| | MAE | F1_Score | MAE | F1_Score |
|---|---|---|---|---|
| MF [19] | 24.17±1.32 | 73.64±0.74 | 23.83±1.25 | 69.01±2.74 |
| NCF [11] | 18.80±0.54 | 89.08±0.07 | 18.86±0.52 | 89.35±0.06 |
| CMF [30] | <u>14.53±1.51</u> | 89.32±0.04 | <u>13.22±0.78</u> | 89.07±0.22 |
| CoNET [12] | 17.46±0.61 | <u>89.59±1.45</u> | 17.18±0.59 | 89.22±0.77 |
| DDTCDR [45] | 20.17±0.56 | 82.60±2.37 | 17.15±0.54 | <u>90.06±0.39</u> |
| Ours | **13.17±0.16** | **90.60±0.37** | **13.01±0.14** | **90.80±0.22** |
| Improvement (%) | 9.36 % | 1.12 % | 1.58 % | 0.57% |
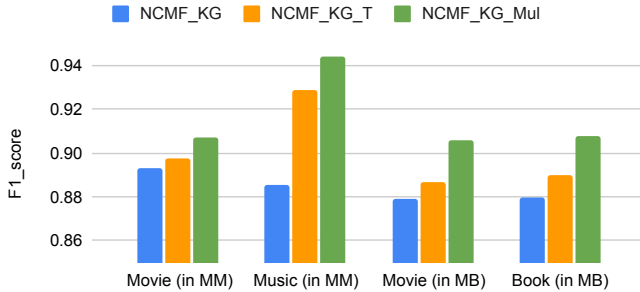


Figure 4: Different ways to incorporate KG information for CDR.

in terms of MAE and F1_score. In particular, our model improves over the strongest baselines $w.r.t.$ MAE by 21%, 15.18% in movie, music (Table 2) respectively, which justifies the effectiveness of our method in integrating items' KG information. If we compare between these two tasks, MM and MB, the improvement on music in MM is more remarkable compared to the performance in MB. Possible reasons are 1) the data is more sparse in the user-music interaction matrix, so leveraging KG information can greatly relieve the sparsity problem (we have verified this in the later experiments:



| (a) Movie (in MM). | (b) Music (in MM). |
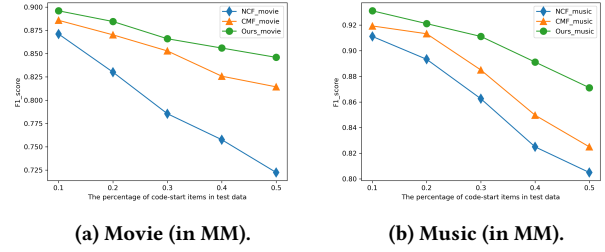
Figure 5: Comparison of different models in cold-start items scenarios.

Comparisions for cold-start item scenarios); 2) the extracted KG contains much useful information, especially for two closely related domains (movie and music both belong to multi-media datasets). Besides, CDR models (CMF,CoNet,DDTCDR) achieve better performance than SDR models (MF, NCF), indicating that utilizing extra information from other resources benefits the performance of recommendation.

## 4.5 Different ways to incorporate KG

We explore different ways to combine item embeddings learned from KG and user-item interaction matrices. NMF_KG takes KG-level embeddings as input, then incorporates them with item embeddings learned from user-item interaction matrices via an aggregation method, e.g., concatenation. NCMF_KG_T tries to refine item embeddings learned from KG with a one-layer MLP and concatenates with embeddings learned from the user-item interaction matrix. NCMF_KG_mul maximizes MI between the two types of representations to guarantee the highly correlated relationship. The results are shown in Fig. 4. Generally, refining the learned KG-level embeddings gets better performance than direct utilization. This is because in real-world KGs (e.g., Freebase) some noises are inevitably introduced in the process of automatically constructing large-scale KGs due to limited labour supervision [14, 38]. NCMF_KG_mul gets the best performance. The possible reason is that item embeddings jointly learn from the user-item rating matrix and entity embeddings from KG, which contain both domain-general and domain-specific knowledge and the neural mutual information estimator can ensure their correlation. Such design is more suitable for the cross-domain recommendation task.

## 4.6 Comparisons for cold-start item scenarios

The KG a natural bridge for items from different domains, which can further alleviate the item cold-start problem in RS.

To validate this, we compare our methods with NCF, CMF under the code-start scenario. We set up the cold-start environment by sampling a subset of items for testing which are unseen in the training data. Results for cold-start items on movie-music datasets are shown in Fig. 5. NCF (the SDR model) is greatly influenced and gets the poorest performance, especially there are a large proportion new items. CMF (the CDR model) can leverage information from two domains, thus it can alleviate the cold-start problem in some extent. Our model goes further to learn representations for cold items from the KG, offering additional information beyond user-item interaction matrices.

# 5 CONCLUSION

In this paper, we constructed a new dataset *AmazonKG4CDR*, the first in the filed linking KG information for cross-domain recommendation. Moreover, we proposed a KG-aware NeuCMF model to learn domain-specific and domain-general knowledge using graph autoencoding strategy to capture both adjacent and higher-order neighborhood information from KG. Our model unified item embeddings learned from user-item interaction matrices and KG with a neural collaborative filtering framework under a mutual information-based neural estimator. Through extensive experiments on real-world datasets, we demonstrated that KG-aware NeuCMF has achieved substantial gains over state-of-the-art baselines. For future work, we will explore the explainability of cross-domain recommendation.

## REFERENCES

[1] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. 2018. Mutual information neural estimation. In *ICML*.
[2] Shlomo Berkovsky, Tsvi Kuflik, and Francesco Ricci. 2007. Cross-domain mediation in collaborative filtering. In *International Conference on User Modeling*. Springer, 355–359.
[3] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*. 1247–1250.
[4] Rose Catherine and William Cohen. 2016. Personalized recommendations using knowledge graphs: A probabilistic logic programming approach. In *Proceedings of the 10th ACM conference on recommender systems*.
[5] Niel Chah. 2017. Freebase-triples: A methodology for processing the freebase data dumps. *arXiv preprint arXiv:1712.08707* (2017).
[6] Lisa Ehrlinger and Wolfram Wöß. 2016. Towards a Definition of Knowledge Graphs. *SEMANTiCS* (2016).
[7] Ignacio Fernández-Tobías and Iván Cantador. 2014. Exploiting Social Tags in Matrix Factorization Models for Cross-domain Collaborative Filtering.. In *CBRecSys@ RecSys*. Citeseer.
[8] Ignacio Fernández-Tobías, Iván Cantador, Marius Kaminskas, and Francesco Ricci. 2012. Cross-domain recommender systems: A survey of the state of the art. In *Spanish conference on information retrieval*. sn, 1–12.
[9] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. 2020. A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering* (2020).
[10] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. 2016. Dual learning for machine translation. *Advances in neural information processing systems* (2016).
[11] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *WWW*.
[12] Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. Conet: Collaborative cross networks for cross-domain recommendation. In *CIKM*.
[13] Liang Hu, Jian Cao, Guandong Xu, Longbing Cao, Zhiping Gu, and Can Zhu. 2013. Personalized recommendation via cross-domain triadic factorization. In *WWW*.
[14] Shengbin Jia, Yang Xiang, Xiaojun Chen, Kun Wang, and E. Shijia. 2019. Triple Trustworthiness Measurement for Knowledge Graph. *WWW* (2019).
[15] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.
[16] Thomas N Kipf and Max Welling. 2016. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308* (2016).
[17] Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.
[18] Thomas N Kipf and Max Welling. 2017. Variational graph auto-encoders. *Advances in Neural Information Processing Systems (workshop)* (2017).
[19] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[20] Pan Li and Alexander Tuzhilin. 2020. DDTCDR: Deep dual transfer cross domain recommendation. In *Proceedings of the 13th International Conference on Web Search and Data Mining*.
[21] Jianxun Lian, Fuzheng Zhang, Xing Xie, and Guangzhong Sun. 2017. CCCFNet: a content-boosted collaborative filtering neural network for cross domain recommender systems. In *Proceedings of the 26th international conference on World Wide Web companion*.
[22] B. Loni, Yue Shi, M. Larson, and A. Hanjalic. 2014. Cross-Domain Collaborative Filtering with Factorization Machines. In *ECIR*.
[23] Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. 2017. Cross-Domain Recommendation: An Embedding and Mapping Approach.. In *IJCAI*.
[24] Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *EMNLP-IJCNLP*.
[25] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22, 10 (2009), 1345–1359.
[26] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2011. Introduction to recommender systems handbook. In *Recommender systems handbook*. Springer, 1–35.
[27] Shaghayegh Sahebi and Peter Brusilovsky. 2015. It Takes Two to Tango: An Exploration of Domain Pairs for Cross-Domain Collaborative Filtering. *Proceedings of the 9th ACM Conference on Recommender Systems* (2015).
[28] Shaghayegh Sahebi and Trevor Walker. 2014. Content-Based Cross-Domain Recommendations Using Segmented Models. In *CBRecSys@RecSys*.
[29] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European semantic web conference*. 593–607.
[30] Ajit P Singh and Geoffrey J Gordon. 2008. Relational learning via collective matrix factorization. In *SIGKDD*.
[31] Xiaoli Tang, Tengyun Wang, Haizhi Yang, and Hengjie Song. 2019. AKUPM: Attention-enhanced knowledge-aware user preference model for recommendation. In *SIGKDD*.
[32] Petar Velickovic, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. 2019. Deep Graph Infomax.. In *ICLR*.
[33] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2018. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *CIKM*. 417–426.
[34] Hongwei Wang, Miao Zhao, Xing Xie, Wenjie Li, and Minyi Guo. 2019. Knowledge graph convolutional networks for recommender systems. In *WWW*.
[35] Pengyang Wang, Yanjie Fu, Yuanchun Zhou, Kunpeng Liu, Xiaolin Li, and Kien Hua. 2020. Exploiting mutual information for substructure-aware graph representation learning. In *IJCAI*.
[36] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering* 29, 12 (2017), 2724–2743.
[37] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. 2019. KGAT: Knowledge Graph Attention Network for Recommendation. In *SIGKDD*.
[38] Ruobing Xie, Zhiyuan Liu, and M. Sun. 2018. Does William Shakespeare REALLY Write Hamlet? Knowledge Representation Learning with Confidence. In *AAAI*.
[39] Keyulu Xu, Chengtao Li, Yonglong Tian, Tomohiro Sonobe, Ken-ichi Kawarabayashi, and Stefanie Jegelka. 2018. Representation Learning on Graphs with Jumping Knowledge Networks. In *ICML*.
[40] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *ICLR*.
[41] Feng Yuan, Lina Yao, and Boualem Benatallah. 2019. DARec: Deep domain adaptation for cross-domain recommendation via transferring rating patterns. In *IJCAI*.
[42] Feng Yuan, Lina Yao, and B. Benatallah. 2019. DARec: Deep Domain Adaptation for Cross-Domain Recommendation via Transferring Rating Patterns. In *IJCAI*.
[43] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2019. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys (CSUR)* 52, 1 (2019), 1–38.
[44] Jun Zhao, Zhou Zhou, Ziyu Guan, Wei Zhao, Wei Ning, Guang Qiu, and Xiaofei He. 2019. Intentgc: a scalable graph convolution framework fusing heterogeneous information for recommendation. In *SIGKDD*.
[45] Feng Zhu, Chaochao Chen, Yan Wang, Guanfeng Liu, and Xiaolin Zheng. 2019. DTCDR: A framework for dual-target cross-domain recommendation. In *CIKM*.
[46] Feng Zhu, Yan Wang, Chaochao Chen, Jun Zhou, Longfei Li, and Guanfeng Liu. 2021. Cross-domain recommendation: challenges, progress, and prospects. In *IJCAI*.