IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 17 May 2022, accepted 20 June 2022, date of publication 5 July 2022, date of current version 13 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3188709

RESEARCH ARTICLE

Learning Domain-Independent Representations via Shared Weight Auto-Encoder for Transfer Learning in Recommender Systems

QINQIN WANG, DIARMUID OŔEILLY-MORGAN, ELIAS Z. TRAGOS, NEIL HURLEY, BARRY SMYTH, AONGHUS LAWLOR, AND RUIHAI DONG

Insight Centre for Data Analytics, University College Dublin, Belfield, Dublin 4, D04 V1W8 Ireland

Corresponding authors: Qinqin Wang (qinqin.wang@insight-centre.org) and Elias Z. Tragos (elias.tragos@insight-centre.org)

This work was supported in part by the Science Foundation Ireland Through the Insight Centre for Data Analytics under Grant SFI/12/RC/2289_P2; and in part by Samsung Electronics Company, Seoul, Republic of Korea.

ABSTRACT Despite many recent advances, state-of-the-art recommender systems still struggle to achieve good performance with sparse datasets. To address the sparsity issue, transfer learning techniques have been investigated for recommender systems, but they tend to impose strict constraints on the content and structure of the data in the source and target domains. For transfer learning methods to work well, there should normally be homogeneity between source and target domains, or a high degree of overlap between the source and target items. In this paper we propose a novel transfer learning framework for mitigating the effects of sparsity and insufficient data. Our method requires neither homogeneity nor overlap between the source and target domains. We describe and evaluate a shared parameter auto-encoder to jointly learn representations of user/item aspects in two domains, applying Maximum Mean Discrepancy (MMD) loss during training to ensure that the source and target representations are similar in the distribution space. The approach is evaluated using a number of benchmark datasets to demonstrate improved recommendation performance when learned representations are used in collaborative filtering. The code used for this work is available on github.com.

INDEX TERMS Recommender system, neural networks, transfer learning, domain adaptation.

I. INTRODUCTION

Collaborative filtering (CF) [15] is a standard recommendation algorithm which harnesses the user-item rating data to construct user and item latent representations. However, the natural sparsity of the user-item rating matrix places limits on the performance of the CF recommendation algorithm. Many recommender approaches exist for improving performance when relationships between users/items can be inferred from side information. Additionally, many transfer learning approaches can exploit similar techniques if a partial mapping can be constructed between user/items in different domains. However, in the absence of side information shared between domains, or an explicit overlap between the domains, there have being few works demonstrating an actual performance bonus from transfer learning in recommender systems [52]. In this work, we outline a method for estimating a mapping between two domains, in the absence of both side information and an overlap, thus allowing for us to exploit such techniques while working only from the raw interaction data.

One common strategy for dealing with sparsity in the rating data is to leverage *side information* to enrich the user-item representations [28], [42], [43]. For example, many researchers use social networks [23], knowledge graphs [40] or multimedia information such as user generated reviews [22] or images [18]. While such approaches have proven to be effective, they are limited to the use of

The associate editor coordinating the review of this manuscript and approving it for publication was Yonghong Peng.

in-domain [23] or homogeneous domain [1] information, i.e. side information must have a certain association with a user or an item [41].

Furthermore, a major challenge in generating informative user/item embeddings or rating patterns from different domains is the problem of learning accurate mapping relations between them [52]. The naive strategy is to directly replace the user/item embeddings in the target domain with those of similar user/items in the source domain [50], or to employ feature combination [32], or active learning [49] to transfer knowledge from similar users across multiple domains. However, identifying similar users/items across different domains is generally a bottleneck in cross-domain transfer learning, especially given that 1. There are often no users/items common to both domains, and 2. Different domain datasets exhibit heterogeneous structures where we can not directly utilise implicit information to build a relation mapping between users/items. For example, there are no overlapping items between the Movielens 100k and Amazon Movie/TV datasets, and it is not clear how to match users between them, i.e. their items and user representations are in independent spaces. An intuitive solution is to project them into a common space in which different domain items (or users) are comparable, while at the same time keeping as much of their own characteristics as possible. The auto-encoder architecture is a common approach to discover an intermediate representation with useful properties. Bengio *et al.* [9] suggest that auto-encoder frameworks are well suited to domain adaptation, as they are able to discover intermediate representations for different domains by working with transfer learning (TL) approaches. In addition, TL approaches aim to apply the results of learning from one domain (the source) to a new domain (the target) [30], [52]. These methods have proven to be effective in computer vision [37], natural language processing [33], and speech recognition settings [5]. The core approach of these TL methods is to fuse multiple datasets from different domains into a similar distribution by adding constraints such as Maximum Mean Discrepancy loss (MMD) [29] or Joint Distribution Adaptation (JDA) [21].

Based on the NU-NI or No User-No Item overlap scenario described by Khan *et al.* [12], we developed our model. The ratings of both domains are analysed to identify the similarity of users and items. However, the bottleneck for the comparison of different domain users and items are the data structure barrier. We can not directly utilise implicit information to build a relation mapping between users/items because of heterogeneous representations from different domains. So we designed the share auto-encoder framework to allow us to project data structures from a variety of domains into a comparable space. This architecture is capable of capturing representations of entities in different domains in such a way that datasets with heterogeneous structures become comparable. With the MMD loss constraint, we can guarantee that the projections of different domains have similar distributions in the kernel space. Second, after extracting embeddings from both domains, we apply a statistical scheme to measure cross-domain sample-wise relevance, that calculates the similarity between each pair of source and target domain samples using cosine similarity, in order to capture the entity relation across different domains. So that we can identify similar items across domains. Next, pre-trained embeddings from the source domain are obtained by Neural Matrix Factorization. Finally, we propose a novel cross-domain regularized matrix factorization method for the target domain task.

Our model is a general and extensible framework that discovers user/item relationships across different domains. This information can then be applied to CF models with a transfer learning regulariser to enhance the recommendations. We apply our framework to four publicly available datasets: MovieLens,[1] Netflix,[2] BookReads,[3] and Amazon Reviews[4] [27]. Our experimental results show that by incorporating learned cross-domain representations, our approach can improve on the performance of a conventional CF model. Additionally, empirical results show that our model is able to boost the entity correlation between two domain datasets, so as to find valid user/item pairs between different domains. The main contributions of the paper are:

- A novel and extensible model is proposed to capture user/item relations between different domains;
- A general framework is proposed for recommendation with cross-domain knowledge;
- A novel measurement is provided to evaluate the cross-domain transfer ability of models in the recommender setting;
- We show that our approach outperforms several state-of-the-art models for both single and cross-domain recommendation, on large-scale real-world datasets.

The rest of the paper is organised as follows. In section II, we give a summary of related works. In section III, we formulate cross-domain adaptation in recommender systems and elaborate our model design. In section IV, we describe the datasets used and the evaluation protocol, then present our experimental results. In section V, we conclude with a discussion of the results and future work.

II. RELATED WORK

To address the sparsity problem, several techniques have been proposed to incorporate side information regarding aspects of users/items, in order to enrich their profiles and generate more informative representations. For example, Ma *et al.* [23] uses social network information to enrich user profiles, and proposes the idea that users with social relationships should be assigned similar user latent factors, exploiting the common real world experience that people generally consult friends for trusted suggestions on movies, books, etc. However, social network information is generally difficult to acquire in real world scenarios. Ideas from transfer learning are also

[1] https://grouplens.org/datasets/movielens/
[2] https://www.kaggle.com/netflix-inc/netflix-prize-data
[3] https://sites.google.com/eng.ucsd.edu/ucsdbookgraph/home
[4] https://nijianmo.github.io/amazon/index.html

used to improve performance in the target domain. Transfer Learning has been exploited to improve CF, by constructing user/item latent factors from information from the source domain as in [31]. This is also done in the TLRec model of Chen *et al.* [1], a framework which uses the user/item-KNN algorithm to find the most similar users/items in the source domain, employing smoothness and regularization to constrain similar user/item latent factors to be close together. Apart from social regularization, side information can also be distilled from text [22] or images [18] to enrich item profiles. Applying zero-shot learning, Li *et al.* [18], learn a projection which maps visual space to semantic space, while [22] design a novel framework based on attention gated recurrent units (GRU) to learn user and item reviews with matrix factorization. By including textual review information, the explainability of their model is significantly improved. Furthermore, [25], [34], [41], [47] leverage user/item attributes, social tags, and browsing histories to create links based on common content. Then, user/item knowledge is transferred across different domains by using this sample-wise relation information. In many cases, these methods are limited by the required homogeneity of the data. Side information must have a certain association with users or items in both domains. In the TL scenario, sample mapping methods are proposed in few-shot learning [6]. In order to alleviate negative transfer at the fine-tuning step, a metric is proposed to measure sample-wise correspondence between two different domains, using the cosine similarity method, so that closely related knowledge should be allotted a higher weighting. The similarity metric is constructed by comparing pair-wise sample embeddings between two domains using cosine similarity. In latent space, Fu *et al.* [7] also propose a MLP-based non-linear mapping function to learn latent space user mapping functions across different domains.

In recommender systems, it is still a challenge to learn accurate cross-domain mapping relations between users/items, though several solutions have previously been proposed. Li, et al propose a solution to the matrix sparsity problem combining datasets from different domains. Their proposed Codebook Transfer Model [16] is based on the rating matrix and jointly clusters user and item representations using an Orthogonal Matrix Tri-Factorisation (ONMTF) [3]. Codebook shows that the rating matrices of different domains can share identical clustered level matrices, which represent the rating patterns of groups of users and items. Extending this method, Li *et al.* presented a more general approach, Rating-Matrix Generative Model (RMGM) [17], which transfers rating knowledge across multiple domains. Also based on Codebook, Cluster-Level Latent Factor Model (CLFM) [8] is proposed to separate common latent patterns and domain specific latent patterns in each domain. By encoding both the common and domain specific patterns in the codebook, the authors show an improvement on the rating prediction task. Furthermore, Zhang, et al propose a pipeline for adaptive knowledge transfer across different domains. Their Consist Information Transfer(CIT) model [48] uses JDA [21] on latent layer domain adaptation between different domains. Inspired by the methods above, [2] makes use of meta-information and labels from the user/item perspective, to cluster them together, with the aim of alleviating data sparsity.[5] The above methods attempt to retrieve additional information from different domains, but ignore any dependency between the source and target domains. The TALMUD (Transfer Learning for Multiple Domains) [26] model takes Codebooks from different source domains independently and aggregates them into a single model. However, rather than using a simple weighted sum, it assigns each Codebook a weight which keeps the difference and independence of the different domains. Notwithstanding, in a later work [4], Cremonesi, et al suggest that a cross-domain CF technique based on co-clustering is not able to transfer knowledge between non-overlapping domains, and they demonstrate this by showing that a randomly initialized Codebook performs just as well as the trained algorithm.

Most of the latent factor-based transfer learning algorithms require some overlap between users and items in datasets. They utilise overlapping content, either side information or user/item profile based [19], [24], to build correlations across different domains. However, these methods impose a strict constraint on the choice of datasets used. For our proposed method, we are mainly focusing on locating the valid neighbours of each item in the target domain from the source domain. We do not require homogeneity between the source domain and target domain, which enlarges the space of possible source domains. To improve the performance of recommendations in the target domain, we use the neighbour (source domain) information.

III. PROPOSED METHOD

A. NOTATION AND PROBLEM FORMULATION

In a transfer learning setting, there are two domains, the "target domain", denoted by D_{Tgt} and the "source domain" denoted by D_{Src}, such that $D_{\mathrm{Src}} \neq D_{\mathrm{Tgt}}$. Given a learning task on D_{Tgt}, the aim is to use knowledge from the source domain to improve the target domain prediction function $f_{\mathrm{Tgt}}(.)$ [45]. We aim to apply transfer learning in the context of recommendation by exploiting knowledge from a source domain recommendation context, to support the learning of a recommendation model in a target domain. Importantly, and in contrast to other research, we do not assume that there is any overlap between either the users or the items in each domain (this is the NU-NI or No User–No Item overlap scenario in the language of Khan *et al.* [13]). We use $\mathcal{U}$ and $\mathcal{I}$ to represent sets of users and items, respectively. In particular, we use $\mathcal{U}_{\mathrm{Src}}$ and $\mathcal{I}_{\mathrm{Src}}$ for source domain users and items, and similarly, $\mathcal{U}_{\mathrm{Tgt}}$ and $\mathcal{I}_{\mathrm{Tgt}}$ for target domain users and items. In general $\mathcal{U}_{\mathrm{Src}} \cap \mathcal{U}_{\mathrm{Tgt}} = \emptyset$ and $\mathcal{I}_{\mathrm{Src}} \cap \mathcal{I}_{\mathrm{Tgt}} = \emptyset$. We apply our model to a rating prediction task, where the goal is to predict, given $u \in \mathcal{U}_{\mathrm{Tgt}}$ and $i \in \mathcal{I}_{\mathrm{Tgt}}$, the rating that user u would

[5] It refers to the difficulty in finding sufficient reliable similar users since in general the active users only rated a small portion of items.

give to item i. We use R_{ui} for the actual rating and $\hat{R}_{ui}$ for the predicted rating. Given a test set $T_{test} \subseteq D_{\mathrm{Tgt}}$ of (u, i) pairs from D_{Tgt}, we evaluate the quality of the model using the root mean squared error (RMSE).

B. RECOMMENDATION MODEL

We focus on a MF model for recommendation, in which user and item factors are learned for each user and item in the domain. In particular, given some latent space dimension $k \ll |U|, |I|$, the model learns a k-dimensional vector $\mathbf{u}_u$ associated with each user $u \in \mathcal{U}$ and a k-dimensional vector $\mathbf{v}_i$ for each $i \in \mathcal{I}$. The prediction function then is $\hat{R}_{ui} = \mathbf{u}_u \cdot \mathbf{v}_i$, the inner product of the factors. Let U and V represent the user and item matrices whose rows are the latent vectors. Given a training set T_{train} of ratings R_{ui}, the standard MF model is learned through the minimisation of the following regularised loss function, where η_1 and η_2 are regularisation weights and $\|\cdot\|_F$ is the Frobenius norm:

$$\mathrm{U}, \mathrm{V} = \arg\min_{U,V} \frac{1}{2} \sum_{(u,i)\in T_{train}} (R_{ui} - \mathbf{u}_u \cdot \mathbf{v}_i)^2 + \eta_1 \|\mathrm{U}\|_F^2 + \eta_2 \|\mathrm{V}\|_F^2 \quad (1)$$

1) SIMILARITY BASED REGULARISATION

Several state of the art works have explored methods to extend Eqn. 1 in order to encourage the latent representations of users or items to be as close as possible when there exists a relationship or similarity between them [20], [23], [51]. This notion of similarity is obtained from an external source, rather than directly from the rating data. Hao Ma *et al.* proposed a CF model with social regularization in [23], by adding a regularization to the loss function, aiming to constrain users in a social neighbourhood to have similar user latent vectors. Thus, given a set N_u of neighbours of user u in the social network, the regularisation term $\beta \sum_u \frac{1}{|N_u|} \sum_{v\in N_u} \|\mathbf{u}_u - \mathbf{u}_v\|^2$ is added to Eqn. 1. In this way, the performance of the CF model is improved by adding more social information. Similarly, in [51] in order to take account of tagging information, a user-user similarity s_{uv} based on tags is constructed and the term $\beta \sum_{u,v} s_{uv} \|\mathbf{u}_u - \mathbf{u}_v\|^2$ is added to Eqn. 1.

This idea has been extended to domain adaptation in recommender systems [1], [10]. However, data homogeneity constrains the generality of these models, since they require the shape of the source and the target domains to be identical [1]. Users and items must have an overlap in the two domains [1], [10], otherwise, it is hard to compare users and items cross different domains.

C. THE AUTO-SHARE MODEL

In this paper, we adapt the idea of regularising the MF model, as described in the previous section. Given some item embeddings, $\mathbf{v}_i^{(\mathrm{Src})}$ learned from the source domain, we learn an MF model in the target domain, regularised so that the item factors of the matrix factorisation are close to the source domain embeddings. In particular, we optimise the following function, where N_i represents some set of nearest-neighbour items to item i in the source domain embedding and β is a hyper-parameter that controls how much information transferred from source domain.

$$\mathrm{U}^{(\mathrm{Tgt})}, \mathrm{V}^{(\mathrm{Tgt})} = \arg\min_{U,V} \frac{1}{2} \sum_{(u,i)\in T_{train}} (R_{ui} - \mathbf{u}_u \cdot \mathbf{v}_i)^2 + \beta \sum_i \frac{1}{|N_i|} \sum_{j\in N_i} \|\mathbf{v}_i - \mathbf{v}_j^{(\mathrm{Src})}\|^2 . \quad (2)$$

The question then is, how can useful item embeddings $\mathbf{v}_i^{(\mathrm{Src})}$ be extracted from the source domain? To answer this question, we appeal to auto-encoders:

1) AUTO-SHARE THROUGH AUTO-ENCODING

With the development of deep learning techniques, auto-encoders have been widely used in domain adaptation [9]. In recommender systems, auto-encoders structures are also applied in both rating prediction and top-N recommendation [14], [35], [44] which get outstanding results on different datasets. A characteristic of auto-encoders is that they learn the data representations efficiently in an unsupervised manner. By modifying the auto-encoder framework, we propose the "Auto-Share model" with Maximum Mean Discrepancy (MMD) loss [29] to address domain adaptation in different domains.

In Fig 1, we propose our novel Auto-Share model which is used to capture entity relations in different domains. Let $\mathbf{R}_i^{(\mathrm{Src})} = (R_{1i}^{(\mathrm{Src})}, R_{2i}^{(\mathrm{Src})}, \ldots R_{mi}^{(\mathrm{Src})}) \in \mathbb{R}^m$, where $m = |\mathcal{U}_{\mathrm{Src}}|$ be a partially observed rating vector of item i in the source domain and $\mathbf{R}_j^{(\mathrm{Tgt})} = (R_{1j}^{(\mathrm{Tgt})}, R_{2j}^{(\mathrm{Tgt})}, \ldots R_{nj}^{(\mathrm{Tgt})}) \in \mathbb{R}^n$, where $n = |\mathcal{U}_{\mathrm{Tgt}}|$ be a partially observed rating vector of item j in the target domain, where $m \neq n$. The goal of this work is to use the auto-encoder model to project both the source domain rating pattern and the target domain rating pattern into the same space, where they are comparable and thus it is possible to find similarities within items of both domains.

We apply the item-based autoRec [35] model to encode these partially observed item vectors. Given a set X of rating vectors in $\mathbb{R}^m$, and a reconstruction function $h(\mathbf{r}; \theta)$, depending on the set of parameters θ, the auto-encoder solves the following minimisation problem:

$$\min_{\theta_S} \sum_{\mathbf{r}\in\mathbf{X}} \|\mathbf{r} - h(\mathbf{r}; \theta)\|^2$$

In fact, we learn two reconstruction functions $h_S(\mathbf{r}; \theta_S)$ and $h_T(\mathbf{r}; \theta_T)$, for the source and target domains simultaneously, with weights shared between the models.

In particular, a two-layer encoder is applied, where the weights of the second layer are shared between the domains. Hence the source domain encoder is

$$Encode_{\mathrm{Src}}(\mathbf{R}^{(\mathrm{Src})}) \equiv \mathbf{E}_{\mathrm{Src}} = g(\mathbf{V}_{share} g(\mathbf{V}_{\mathrm{Src}} \mathbf{R}^{(\mathrm{Src})} + \mu_{\mathrm{Src}}) + \mu_{share})$$

where $g(.)$ is a sigmoid activation function. Similarly for the target domain encoder $Encode_T$. Decoding is also carried out

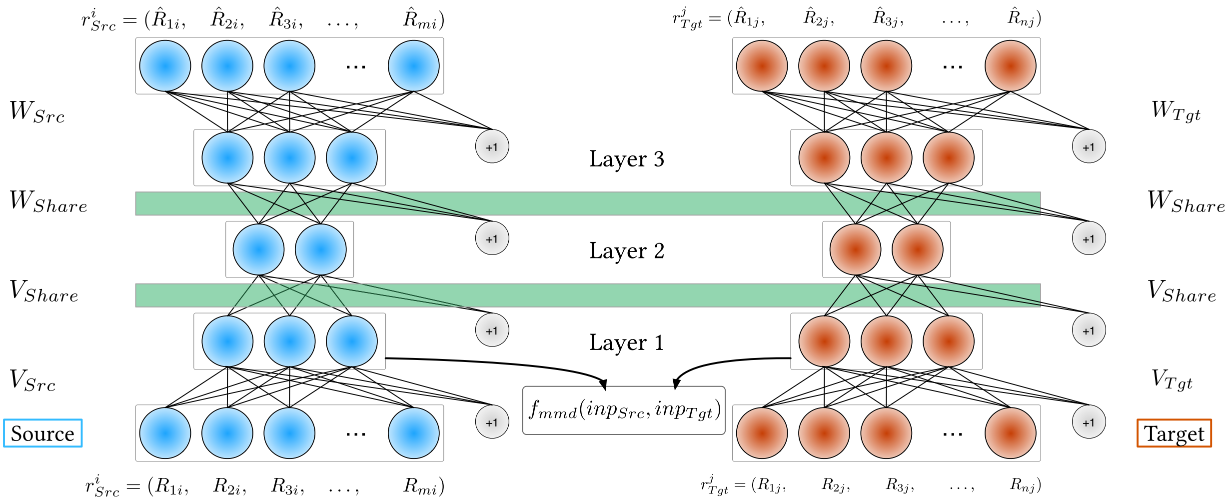

FIGURE 1. Overview of the architecture of Auto-Share. Two auto-encoder frameworks take partially observed rating vectors from different domains. The weights of the second layer are shared between the two domains. MMD loss is added on the first layer, and for learning the cross domain relation we evaluate the embedding on layer 3 where it delivers the best performance.

over two layers where

$$Decode_{\mathrm{Src}}(\mathbf{E}_{\mathrm{Src}}) = f(\mathbf{W}_{\mathrm{Src}} f(\mathbf{W}_{share}\mathbf{E}_{\mathrm{Src}} + b_{share}) + b_{\mathrm{Src}}).$$

where $f(.)$ is the *identity* function. Similarly for the target domain decoder $Decode_T$. Hence, $\hat{\mathbf{R}}^{(\mathrm{Src})} \equiv h_{\mathrm{Src}}(\mathbf{R}^{(\mathrm{Src})}; \theta_{\mathrm{Src}}) = Decode_{\mathrm{Src}}(Encode_{\mathrm{Src}}(\mathbf{R}^{(\mathrm{Src})}))$ and similar to the target domain. The model therefore has the full set of parameters

$$\theta = (\mathbf{V}_{\mathrm{Src}}, \mathbf{V}_{\mathrm{Tgt}}, \mathbf{V}_{share}\mathbf{W}_{\mathrm{Src}}, \mathbf{W}_{\mathrm{Tgt}}, \mathbf{W}_{share}, \mu_{\mathrm{Src}}, \mu_{\mathrm{Tgt}}, \mu_{share}, b_{\mathrm{Src}}, b_{\mathrm{Tgt}}, b_{share}).$$

$\mathbf{V}_{\mathrm{Src}}$ is $m \times k$, $\mathbf{V}_{\mathrm{Tgt}}$ is $n \times k$, $\mathbf{V}_{share}$ is $k \times \ell$, $\mathbf{W}_{\mathrm{Src}}$ is $k \times m$, $\mathbf{W}_{\mathrm{Tgt}}$ is $k \times n$ and $\mathbf{W}_{share}$ is $\ell \times k$, where k and ℓ are hyper-parameters that depend on the size of the datasets on both domains. In our experiments, we take $k = 500$ and $\ell = 200$ in five datasets which give the best performance on learning the relationship between source and target domains.

Additionally, we add the MMD loss which is a kernel-based distance function on the first layer between the two domain auto-encoders. The objective of the MMD loss is to learn some valid elements across domains in a Reproducing Kernel Hilbert Space (RKHS). In the subspace of these elements, different domain representations also have a similar distribution. In summary, we train the Auto-Share model so that it minimises the following loss function:

$$L = \sum_{i \in \mathcal{I}_{\mathrm{Tgt}}} \|\hat{\mathbf{R}}_i^{(\mathrm{Tgt})} - \mathbf{R}_i^{(\mathrm{Tgt})}\|^2 + \sum_{j \in \mathcal{I}_{\mathrm{Src}}} \|\hat{\mathbf{R}}_j^{(\mathrm{Src})} - \mathbf{R}_j^{(\mathrm{Src})}\|^2 + Loss_{MMD}(H_c^{\mathrm{Src}}, H_c^{\mathrm{Tgt}}) + \frac{\alpha}{2}(\|\mathbf{W}_{\mathrm{Src}}\|_F^2 + \|\mathbf{V}_{\mathrm{Src}}\|_F^2 + \|\mathbf{W}_{\mathrm{Tgt}}\|_F^2 + \|\mathbf{V}_{\mathrm{Tgt}}\|_F^2 + \|\mathbf{V}_{share}\|_F^2 + \|\mathbf{W}_{share}\|_F^2)$$

where H_c^{Src} and H_c^{Tgt} are the first encoded representation of the source samples and the first encoded representation of the target samples respectively. To prevent overfitting, regularization of the learned parameters is added in the loss function controlled by the weight α which is 0.1 in our experiments. The aim of this model is to project the source domain and target domain rating patterns into a common space where they can be compared with each other. There are two encoded layers (layer 1, layer 2) and one decoded layer (layer 3) which have the same shape on both domain representations. The Layer 1 is selected to construct MMD loss and Layer 3 is chosen as our comparable space which has the best performance, showed in Table 2 and Table 5.

2) TRAINING THE MF MODEL

Once $\mathbf{v}_j^{(\mathrm{Src})}$ are chosen from an appropriate layer of the auto-encoder, the model of Eqn. 2 can be trained. During the training process, the source domain vectors $\mathbf{v}_j^{(\mathrm{Src})}$ are frozen. To avoid negative transference in the model, we set a high similarity threshold for choosing nearest-neighbour items between the two domains. Only similar items should be incorporated into the training, so not all item latent vectors are used for the target task. The regularisation parameter β controls how much knowledge should be used from the source domain. If $\beta = +\infty$, then the item latent vector $\mathbf{v}_i^{(\mathrm{Tgt})}$ in the target domain is the average of all the nearest neighbour latent vectors in the source domain. Source domain knowledge will not affect the target domain performance if $\beta = 0$.

TABLE 1. Description of the datasets.

dataset	Interaction	Users	Items	Density
ML100K	95,566	943	1,236	0.082
ML1M	472,292	6,037	1,236	0.063
Amazon Movies and TV [27]	370,549	6,833	5.929	0.00916
Netflix	79,361	4,285	4,488	0.00412
Children Book [38], [39]	1,537,605	9,084	6,615	0.0256
Poetry Book [38], [39]	144,938	4,524	1,482	0.021617

TABLE 2. Average rank compare with other methods.

Methods	Average Rank
Random	782
Average Rating	357
Entropy	535
Jensen–Shannon divergence	232
Auto-Share (Layer 1)	183
Auto-Share (Layer 2)	141
Auto-Share (Layer 3)	**134**

IV. EXPERIMENTS

In this section, we investigate how our Auto-Share model transfers knowledge from source to target domain, and how this transfer positively impacts the performance of a target domain CF system. First, we examine item similarities in the shared space learned by the Auto-Share model, and then compare the Auto-Share regularised MF system with two in-domain standard baselines: MF and FMM, and we also compare with the transfer learning models: CodeBook and RMGM. Furthermore, we use the randomised method conducted in [4] where we replace pre-trained embeddings from the source domain by randomly generated embeddings to verify the positive impact of domain transfer on the performance of the target CF system.

A. EXPERIMENT METRICS

We have modelled the recommendation problem at hand as a rating prediction problem. The metric we choose for the evaluation is the Root Mean Square Error (RMSE), as it is the common metric for rating prediction tasks. The evaluation method defined as:

$$\mathrm{RMSE} = \sqrt{\frac{1}{|\hat{R}|} \sum_{\hat{r}_{ui} \in \hat{R}} \left(r_{ui} - \hat{r}_{ui}\right)^2} \tag{3}$$

where $|\hat{R}|$ is the number of prediction during in evaluation.

B. DATASET DESCRIPTION

We use six datasets for evaluation: the ML1M and Children's Book datasets are chosen as source domain datasets, as they have high densities, thus containing more information. Amazon TV, Poetry and Book, ML100k, and Netflix, are chosen as target domains. The dataset statistics are summarised in Table 1. For the Movielens datasets, we choose only the items which overlap between the two domains for the evaluation of average rank shown in Table 2. For Amazon TV dataset, We adopt the commonly applied k-core pruning method as in [7], [11] to filter short profiles where it is filtered by selecting items with at least 50 ratings, and users with at least 20. Netflix is filtered by selecting items with at least 15 ratings and users with at least 5. Both book datasets are filtered by choosing items with at least 100 items. Then we split the datasets for training, testing and validation on both source and target domain with the ratio of 80%, 10% and 10% respectively. We train the models on the train split, tune model hyper parameters on validation split, and report the final results on on the test split.

C. EMBEDDING SPACE CORRESPONDENCES

Using the ML datasets, we examine whether Auto-share can form useful correspondences between items in the source domain and items in the target domain. There are 1,236 movies in common between the ML100k and ML1M datasets. We expect that the source domain embedding of each movie should be close to its target domain embedding in the embedding layers of Auto-share. Note that ML100K was collected between 1997 and 1998, while ML1M contains users who joined MovieLens starting from the year 2000. Thus, for each movie, the rating profiles in each dataset do not have any ratings in common. We run the Auto-share model on randomly selected batches of profiles from the source and target domains. After the model has converged, we examine an Auto-share embedding layer. For each common item, we use the cosine similarity to calculate the distance of its target embedding to the embeddings of all source domain items. We sort the source domain items by this similarity. Ideally the top-ranked source embedding will be that of the item itself. We record its rank in this ordering. The average rank over all items (a value between 1 and the number of items, with low values being better) gives a measure of the quality of the embedding space for finding correct correspondences between the domains, and helps to verify the ability of the model to project datasets from different domains into a comparable space.

Alternatively it is possible to find correspondences between items based on statistics of the item profiles. For example, for each target item, we rank the source items by calculating the L_1 norm between the target's average rating and the average source profile rating. We also rank source items according to the absolute difference between the entropy of the target profile and the entropy of the source profile. Furthermore, the Jensen–Shannon distance (JSD) is a method of measuring the similarity between two probability distributions,

$$\mathrm{JSD}(P\|Q) = \sqrt{\frac{D(p\|m) + D(q\|m)}{2}} \tag{4}$$

where p and q in Eqn. 4 are the probability vectors standing for the rating distribution in the item profile and $m = \frac{1}{2}(p+q)$. For each item in the target domain, we rank the source items by calculating the JSD based on the target item's profile rating distribution and the source's item distribution. The results are shown in Table 2, where we compare the average rank for the two Movielens datasets of the Auto-Share model with three statistical methods. Three different Auto-share layers are examined. The results of Table 2 indicates only based on item profile information, we can hardly form useful correspondences between items in the source domain and item in the target domain. Based on item profile rating distribution, using JSD similarity can reach a promising result which is 232 on average item rank between two domains, but our proposed auto-share method with pre-trained embedding can achieve better performance which is 183, 141, and 134 average item rank using layer 1, layer 2 and layer 3 embedding respectively. In conclusion, Auto-share out-performs the statistical methods, with Layer 3 delivering the best performance.

TABLE 3. Pearson correlation on items between two domains.

Method	Entropy	Mean	Std
Original Item Profile	0.36	0.83	0.26
Item Embedding	**0.88**	**0.90**	**0.90**

From a statistical perspective, the value of cosine similarity between two vectors is associated with the correlation with respect to mean, standard deviation and entropy, with a similar mean, standard deviation and entropy between two vectors leading to a higher cosine similarity. After projecting the original incomparable rating sequences from different domains into the same latent space, we can directly use cosine similarity on latent embeddings to build correlations between items pairs across two domains. To evaluate the correlation difference between the original representations (rating sequences) and the new projected representations, we first calculate the identical item representation's mean, standard deviation and entropy across different domains. Next we analyse the Pearson correlation for identical items using different statistic values across two domains. Table 3 illustrates that after using Auto-share to map different domain item representations into the same space, the item correlation between two different domains improves significantly in term of these three statistic values: mean, standard deviation and entropy. Further experiments are conducted in our ablation study in section IV-F, which verifies the impact of each component of the Auto-share model.

D. COMPETITORS

In this section, we will introduce the state-of-the-art baseline models with which we compare our method. We consider two classic in-domain models, and three cross-domain models where they both do not require users or items overlapping cross two domains and they focus on rating prediction task. Furthermore, since most of recent work requires overlapping on users or items, two state-of-the-art models EMCDR and DDTCDR are chosen to compare with on ML datasets where they have overlapping on items. Last, we design a control method to verify the affect of knowledge transfer from the source domain.

- **CF [15]** Collaborative Filtering (CF) is a classical method that employs matrix factorisation to learn latent factors for users and items.
- **FMM [36]** Flexible Mixture Model (FMM) extends partitioning/clustering algorithms for collaborative filtering by clustering both users and items together without assuming each user/item should belong to a single cluster. It is the basis of cross-domain CF techniques based on co-clustering.
- **Codebook [16]** Codebook is based on the rating matrix, and jointly clusters user and item representations using an Orthogonal Matrix Tri-Factorisation (ONMTF) [3]. Codebook shows that the rating matrices of different domains can share identical clustered level matrices,

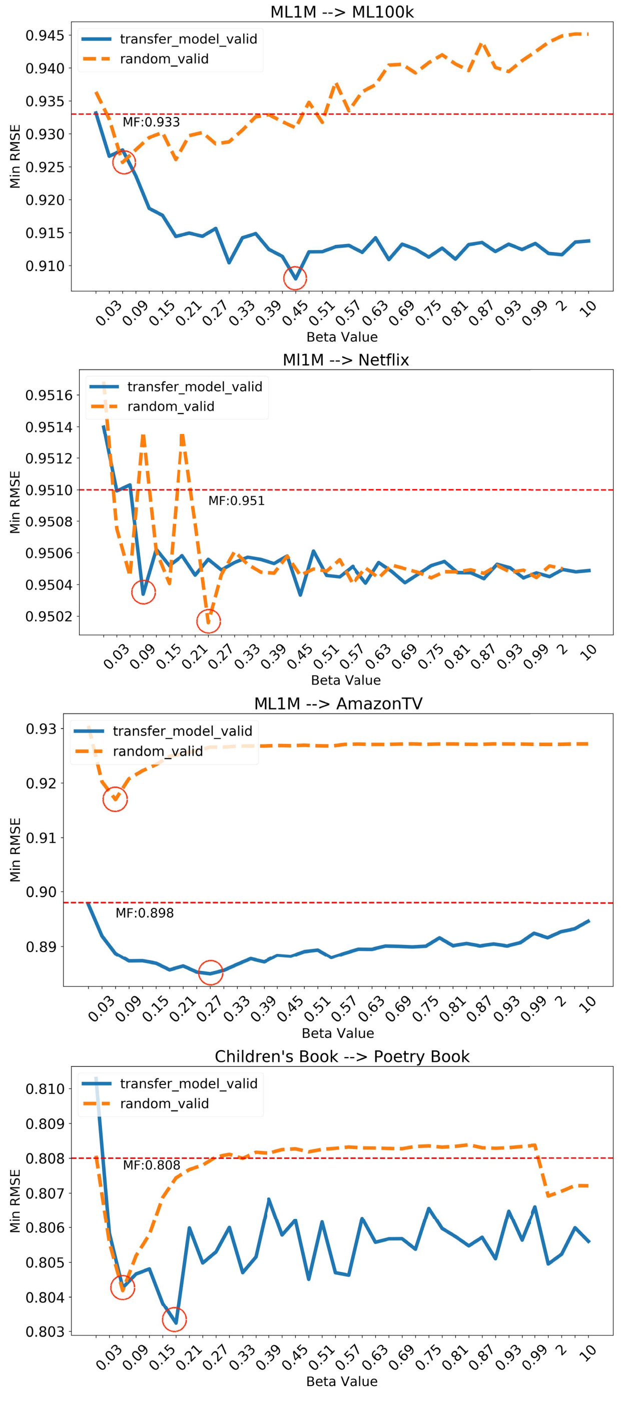

FIGURE 2. Performance analysis for different beta values on different datasets. MF indicates the RMSE of standard Matrix Factorization in the target domain.

TABLE 4. Performance comparison of CF, FMM, RMGM, CodeBook, LKT-FM, DDTCDR, EMCDR and Auto-Transfer.

Model	ML1M to ML100K	ML1M to Amazon TV	Children's Book to Poetry Book	ML1M to Netflix
CF	0.9222	0.8957	0.8038	0.9814
FMM	1.0613	1.1873	0.9835	1.1413
RMGM	0.9811	1.4912	0.9120	1.0021
CodeBook	1.1310	1.1690	0.9535	1.1401
LKT-FM	0.9048	0.8894	0.8134	0.9851
Auto-Transfer with random Emd	0.9183	0.9158	0.7977	0.9816
DDTCDR	0.9425	—	—	—
EMCDR	0.9927	—	—	—
Auto-Transfer	**0.8969**	**0.8858**	**0.7904**	**0.9809**

which can represent the rating patterns of groups of users and items.

- **RMGM [17]** RMGM extends the Codebook model, which transfers knowledge from source to target domain. RMGM is designed to transfer knowledge from multiple source domains into a single target domain.
- **LKT-FM [46]** Low-rank knowledge transfer via factorization machine (LKT-FM) is able to discover high quality knowledge from large and sparse source domain matrices, and to integrate the knowledge without losing much information contained in the target matrix via exploiting Factorization Machine.
- **Auto-Transfer with random embeddings** We propose this method as a control to verify the positive transfer of knowledge from source to target domain. We replace the nearest neighbour embeddings chosen from the source domain with randomly generated embeddings. By comparing Auto-Transfer with this control method, we seek to verify its effectiveness on the target domain task.
- **EMCDR[6] [24]** Embedding and Mapping framework for Cross-Domain Recommendation (EMCDR) proposed the framework to utilise common users or items knowledge to build a map function so as to target to data sparsity problem in target domain.
- **DDTCDR[7] [19]** Deep Dual Transfer Cross Domain Recommendation (DDTCDR) developed a novel latent orthogonal mapping function to extract user preferences over multiple domains while preserving relations between users across different latent spaces.

E. AUTO-SHARE REGULARISED MF MODEL EVALUATION

Employing two datasets as source domain data, we tune our end to end model using the validation data, and evaluate the performance of the Auto-share regularised MF model on holdout data. The β values grid searched for different datasets are summarised in Fig 2. We set the range of β values from 0 to 10. From 0 to 1, we use a step size of 0.03 and the values 2, 5, 10 are searched afterwards. To select the proper MMD loss layer and comparable space, we enumerate different layers. According to Table 5, the MMD loss is constructed on the first layer and the third layer as the comparable space performs the best on the regularised MF model.

To evaluate the impact of pre-trained embeddings from the source domain on the target task, we compare the performance of pre-trained embeddings with randomly generated embeddings using the randomisation method proposed in [4]. Since randomly generated embeddings do not contain any information from the source domain, the optimal β values for randomly generated embedding models are small across all datasets, while the optimal β values for pre-trained embeddings vary across datasets. In Fig 2, we use a red circle to mark the best validation β, and optimal RMSE on the test set is summarised in Table 4 and compared with two in domain methods (standard MF and FMM), and with three cross-domain methods without overlapping on users or items (Codebook, RMGM and LKT-FM). For all baseline methods, implementations are either from open-source project or the original authors. Since the datasets we use are identical to the original method(ML dataset), the parameters are chosen based on those reported in [16], [17], [19], [24], [36], [46]. The parameter β in Eqn. 2 controls how much information we incorporate from the source domain. Despite the fact that the LKT-FM method offers competitive results compared to the Auto-Share model, we must point out that the LKT-FM model is a factorization machine based model, which increases its complexity. Overall Auto-Share model not only outperforms the other state-of-the-art transfer learning methods, but also performs better than a standard MF trained on the target domain. This shows that by incorporating useful information from the source domain using the Auto-share model, performance can be improved over MF algorithms trained solely on the target domain.

F. ABLATION STUDY

1) EFFECT OF SHARED PARAMETERS IN THE AUTO-SHARE MODEL

Although we have demonstrated strong empirical results, the results presented so far have not isolated the specific contributions from each component of the Auto-Share model. In this section, we conduct an ablation study on the Auto-Share model, using the ML datasets, so as to better understand the effectiveness of each model component. Table 6

[6] https://github.com/MaJining92/EMCDR
[7] https://github.com/lpworld/DDTCDR

TABLE 5. Grid search on different layers of MMD loss and embedding layers. The MMD loss is evaluated on cross various auto-encoder layers and different layers are chosen as the comparable space.

MMD Layer	Comparable Space	RMSE
1	1	0.9113
1	2	0.9072
1	**3**	**0.8986**
2	1	0.9193
2	2	0.9189
2	3	0.9238
3	1	0.9257
3	2	0.9203
3	3	0.9194

TABLE 6. Ablation study on Auto-Share model.

Methods	Average Rank	RMSE
Two Independent Auto-Encoders	609	0.9207
Auto-Share without MMD	327	0.9149
Auto-Share with MMD	**134**	**0.8969**
Auto-Share with New Data	**0.5**	–

shows the results of our tests on different versions of the Auto-Encoder model with the same measurement of average rank used in section IV-C and the RMSE on regularised MF model. First, we employ two simple auto-encoder models with one hidden layer [35] to train source domain and target domain datasets independently. After both models converge, we examine the embedding layer using average rank to measure item correspondence between two domains. We find that the average rank on two independent auto-encoders is **609**. For the purpose of regularised MF training, we select relevant items from the pre-trained source domain embedding by the comparable space. We get 0.9207 on RMSE which is similar to the random embedding regularised. Then, we change the framework into shared-parameter auto-encoders with two hidden layers, the average rank decreases to **327**. Comparable space between the source domain and target domain are set on layer 3 where we get 0.9149 RMSE on regularised MF model on the target domain. Finally, by adding the MMD loss on the first layer between the two domain auto-encoders, we get our proposed shared-parameter auto-encoder framework which has the best performance, with average rank of **134** and with RMSE of **0.8969** on target domain. According to the experiments, the Auto-share model can project item rating profiles from two domains into a comparable space, where we can use cosine similarity to capture entity relations across different domains. Exploiting the captured entity relation, the data sparsity problem on the target domain can be mitigated by choosing the nearest neighbors from the source domain.

Furthermore, we seek to explore what features can be captured using a shared parameter auto-encoder on different datasets. Taking the ML1M dataset as a source domain dataset, we create another dataset by randomly permuting the columns of the rating matrix, resulting in a dataset where the item profiles are different, but their rating pattern distributions are the same as the source domain. We use the ''new'' permuted ML1M as the target domain dataset and run Auto-share on randomly selected batches of profiles from the source and target domains. After the model converges, we examine the embedding layer correspondence between two different domains using average rank measurement. The average rank is **0.5** which shows that the Auto-Share model can successfully capture the rating distribution between different domains.

G. LIMITATION

As compared with the state-of-the-art models, the Auto-Share model has demonstrated competitive performance and eases the restriction that requires overlapping users or items cross different domains, but as discussed in this section, this method still has the following limitations: 1) Our experiments are based on domains with explicit user feedback, such as ratings. Despite demonstrating the generality and extensibility of our framework, the stringent requirements of the dataset limit the domain we choose. In real life, most datasets contain only implicit feedback from users, such as whether they like or dislike a specific item. Additionally, for the Click-through rate (CTR) prediction task, the domain only contains users' browsing history, which indicates whether a user clicked on the item or not. To date, we have not incorporated the aforementioned data into our model, but we plan to develop a framework for fitting different kinds of data in the future. 2) For embedding correspondence evaluation, we presume items across different domains should have a similar profile, so their presentations in different domains should be consistent. This assumption no longer holds true when domains are so different. In Europe, for instance, some books are popular, but are less so in Asia. Therefore, we must carefully define our source domain and target domain. 3) Last but not least, the auto-encoder model will not be able to locate the corresponding items across two different domains accurately if the item profile only has a limited number of interactions, which will introduce noise during the training process.

V. CONCLUSION

We have demonstrated the Auto-Share model that can learn relations between items in non-overlapping domains, in the absence of side information, and shown that in combination with regularisation techniques, it is possible to exploit these relations to increase recommendation performance. Contrary to many existing approaches, we depend only on the raw interaction data. We have proposed a novel and extensible deep shared-weight auto-encoder method to extract relational knowledge from different domains. We have demonstrated experimentally that by applying information learned from the source domain to the target domain, we can positively alleviate the sparsity problem for collaborative filtering recommendation models. Furthermore, by an empirical comparison with randomly generated embeddings, we have shown that our model effectively performs a positive transfer of knowledge from the source domain to the target domain. Last but not least, this framework helps to remove limitations on the usability of source domains. We have shown that, in the absence of side information, and even if the source domain and the target domains do not have overlapping users/items, some knowledge is still transferable.

REFERENCES

[1] L. Chen, J. Zheng, M. Gao, A. Zhou, W. Zeng, and H. Chen, "TLRec: Transfer learning for cross-domain recommendation," in *Proc. IEEE Int. Conf. Big Knowl. (ICBK)*. Washington, DC, USA: IEEE Comput. Soc., Aug. 2017, pp. 167–172.

[2] W. Chen, W. Hsu, and M. L. Lee, "Making recommendations from multiple domains," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, I. S. Dhillon, Y. Koren, R. Ghani, T. E. Senator, P. Bradley, R. Parekh, J. He, R. L. Grossman, and R. Uthurusamy, Eds. Chicago, IL, USA: ACM, Aug. 2013, pp. 892–900, doi: 10.1145/2487575.2487638.

[3] T. L. C. Ding and H. P. W. Peng, "Orthogonal nonnegative matrix tri-factorizations for clustering," in *Proc. KDD*, Philadelphia, PA, USA, Aug. 2006, pp. 126–135.

[4] P. Cremonesi and M. Quadrana, "Cross-domain recommendations without overlapping data: Myth or reality?" in *Proc. 8th ACM Conf. Recommender Syst. (RecSys)*, 2014, pp. 297–300.

[5] J. Deng, Z. Zhang, E. Marchi, and B. Schuller, "Sparse autoencoder-based feature transfer learning for speech emotion recognition," in *Proc. Humaine Assoc. Conf. Affect. Comput. Intell. Interact.* Washington, DC, USA: IEEE Comput. Soc., Sep. 2013, pp. 511–516.

[6] S. Feng and M. F. Duarte, "Few-shot learning-based human activity recognition," *CoRR*, vol. abs/1903.10416, pp. 1–29, Mar. 2019.

[7] W. Fu, Z. Peng, S. Wang, Y. Xu, and J. Li, "Deeply fusing reviews and contents for cold start users in cross-domain recommendation systems," in *Proc. 33rd AAAI Conf. Artif. Intell., 31st Innov. Appl. Artif. Intell. Conf., 9th AAAI Symp. Educ. Adv. Artif. Intell.* Honolulu, HI, USA: AAAI Press, Jan. 2019, pp. 94–101, doi: 10.1609/AAAI.V33I01.330194.

[8] S. Gao, H. Luo, D. Chen, S. Li, P. Gallinari, and J. Guo, "Cross-domain recommendation via cluster-level latent factor model," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, in Lecture Notes in Computer Science, vol. 8189. Berlin, Germany: Springer, 2013, pp. 161–176.

[9] X. Glorot, A. Bordes, and Y. Bengio, "Domain adaptation for large-scale sentiment classification: A deep learning approach," in *Proc. ICML*. Madison, WI, USA: Omnipress, 2011, pp. 513–520.

[10] M. Jiang, P. Cui, N. J. Yuan, X. Xie, and S. Yang, "Little is much: Bridging cross-platform behaviors through overlapped crowds," in *Proc. AAAI*. Palo Alto, CA, USA: AAAI Press, 2016, pp. 13–19.

[11] S. Kang, J. Hwang, D. Lee, and H. Yu, "Semi-supervised learning for cross-domain recommendation to cold-start users," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.* W. Zhu, D. Tao, X. Cheng, P. Cui, E. A. Rundensteiner, D. Carmel, Q. He, and J. X. Yu, Eds. Beijing, China: ACM, Nov. 2019, pp. 1563–1572, doi: 10.1145/3357384.3357914.

[12] M. M. Khan, R. Ibrahim, and I. Ghani, "Cross domain recommender systems: A systematic literature review," *ACM Comput. Surv.*, vol. 50, no. 3, pp. 36:1–36:34, 2017, doi: 10.1145/3073565.

[13] M. M. Khan, R. Ibrahim, and I. Ghani, "Cross domain recommender systems: A systematic literature review," *ACM Comput. Surv.*, vol. 50, no. 3, pp. 36:1–36:34, 2017.

[14] Z. A. Khan, S. Zubair, K. Imran, R. Ahmad, S. A. Butt, and N. I. Chaudhary, "A new users rating-trend based collaborative denoising auto-encoder for top-N recommender systems," *IEEE Access*, vol. 7, pp. 141287–141310, 2019.

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

[16] B. Li, Q. Yang, and X. Xue, "Can movies and books collaborate? Cross-domain collaborative filtering for sparsity reduction," in *Proc. 21st Int. Joint Conf. Artif. Intell.*, C. Boutilier, Ed. Pasadena, CA, USA, Jul. 2009, pp. 2052–2057. [Online]. Available: http://ijcai.org/Proceedings/09/Papers/338.pdf

[17] B. Li, Q. Yang, and X. Xue, "Transfer learning for collaborative filtering via a rating-matrix generative model," in *Proc. 26th Annu. Int. Conf. Mach. Learn.*, in ACM International Conference Proceeding Series, vol. 382, Jun. 2009, pp. 617–624.

[18] J. Li, M. Jing, K. Lu, L. Zhu, Y. Yang, and Z. Huang, "From zero-shot learning to cold-start recommendation," in *Proc. AAAI*. Menlo Park, CA, USA: AAAI Press, 2019, pp. 4189–4196.

[19] P. Li and A. Tuzhilin, "DDTCDR: Deep dual transfer cross domain recommendation," in *Proc. 13th Int. Conf. Web Search Data Mining*, J. Caverlee, X. B. Hu, M. Lalmas, and W. Wang, Eds. Houston, TX, USA, Jan. 2020, pp. 331–339, doi: 10.1145/3336191.3371793.

[20] W.-J. Li and D.-Y. Yeung, "Relation regularized matrix factorization," in *Proc. 21st Int. Joint Conf. Artif. Intell.*, 2009, pp. 1126–1131.

[21] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.* Washington, DC, USA: IEEE Comput. Soc., Dec. 2013, pp. 2200–2207.

[22] Y. Lu, R. Dong, and B. Smyth, "Coevolutionary recommendation model: Mutual learning between ratings and reviews," in *Proc. World Wide Web Conf. (WWW)*, Geneva, Switzerland, 2018, pp. 773–782, doi: 10.1145/3178876.3186158.

[23] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *Proc. 4th ACM Int. Conf. Web Search Data Mining (WSDM)*, 2011, pp. 287–296.

[24] T. Man, H. Shen, X. Jin, and X. Cheng, "Cross-domain recommendation: An embedding and mapping approach," in *Proc. 26th Int. Joint Conf. Artif. Intell.*, C. Sierra, Ed. Melbourne, VI, Australia, Aug. 2017, pp. 2464–2470, doi: 10.24963/IJCAI.2017/343.

[25] J. Manotumruksa, D. Rafailidis, C. Macdonald, and I. Ounis, "On cross-domain transfer in venue recommendation," in *Proc. Eur. Conf. Inf. Retr.*, in Lecture Notes in Computer Science, vol. 11437, L. Azzopardi, B. Stein, N. Fuhr, P. Mayr, C. Hauff, and D. Hiemstra, Eds. Cologne, Germany: Springer, 2019, pp. 443–456, doi: 10.1007/978-3-030-15712-8_29.

[26] O. Moreno, B. Shapira, L. Rokach, and G. Shani, "TALMUD: Transfer learning for multiple domains," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, X. Chen, G. Lebanon, H. Wang, and M. J. Zaki, Eds. Maui, HI, USA, Oct./Nov. 2012, pp. 425–434, doi: 10.1145/2396761.2396817.

[27] J. Ni, J. Li, and J. McAuley, "Justifying recommendations using distantly-labeled reviews and fine-grained aspects," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, K. Inui, J. Jiang, V. Ng, and X. Wan, Eds. Hong Kong: Assoc. Comput. Linguistics, Nov. 2019, pp. 188–197, doi: 10.18653/v1/D19-1018.

[28] W. Ouyang, X. Zhang, L. Zhao, J. Luo, Y. Zhang, H. Zou, Z. Liu, and Y. Du, "MiNet: Mixed interest network for cross-domain click-through rate prediction," *CoRR*, vol. abs/2008.02974, pp. 1–8, Aug. 2020.

[29] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011.

[30] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[31] W. Pan, E. W. Xiang, N. N. Liu, and Q. Yang, "Transfer learning in collaborative filtering for sparsity reduction," in *Proc. 24th Conf. Artif. Intell.*, M. Fox and D. Poole, Eds. Atlanta, GA, USA: AAAI Press, Jun. 2010, pp. 230–235. [Online]. Available: http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1649

[32] W. Pan and Q. Yang, "Transfer learning in heterogeneous collaborative filtering domains," *Artif. Intell.*, vol. 197, pp. 39–55, Apr. 2013, doi: 10.1016/j.artint.2013.01.003.

[33] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2018, pp. 1–15.

[34] S. Sahebi and T. Walker, "Content-based cross-domain recommendations using segmented models," in *Proc. 1st Workshop New Trends Content-Based Recommender Syst. Located 8th ACM Conf. Recommender Syst.*, in CEUR Workshop Proceedings, vol. 1245, T. Bogers, M. Koolen, and I. Cantador, Eds. Silicon Valley, CA, USA, Oct. 2014, pp. 57–64. [Online]. Available: http://ceur-ws.org/Vol-1245/cbrecsys2014-paper09.pdf

[35] S. Sedhain, A. K. Menon, S. Sanner, and L. Xie, "AutoRec: Autoencoders meet collaborative filtering," in *Proc. 24th Int. Conf. World Wide Web*, May 2015, pp. 111–112.

[36] L. Si and R. Jin, "Flexible mixture model for collaborative filtering," in *Proc. ICML*. Menlo Park, CA, USA: AAAI Press, 2003, pp. 704–711.
[37] B. Tan, Y. Zhang, S. J. Pan, and Q. Yang, "Distant domain transfer learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1–7.
[38] M. Wan and J. McAuley, "Item recommendation on monotonic behavior chains," in *Proc. 12th ACM Conf. Recommender Syst.*, S. Pera, M. D. Ekstrand, X. Amatriain, and J. O'Donovan, Eds. Vancouver, BC, Canada, Sep. 2018, pp. 86–94, doi: 10.1145/3240323.3240369.
[39] M. Wan, R. Misra, N. Nakashole, and J. McAuley, "Fine-grained spoiler detection from large-scale review corpora," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, A. Korhonen, D. R. Traum, and L. Màrquez, Eds. Florence, Italy: Assoc. Comput. Linguistics, 2019, pp. 2605–2610, doi: 10.18653/V1/P19-1248.
[40] H. Wang, F. Zhang, M. Zhao, W. Li, X. Xie, and M. Guo, "Multi-task feature learning for knowledge graph enhanced recommendation," in *Proc. World Wide Web Conf. (WWW)*, 2019, pp. 2000–2010.
[41] J. Wang and J. Lv, "Tag-informed collaborative topic modeling for cross domain recommendations," *Knowl.-Based Syst.*, vol. 203, Sep. 2020, Art. no. 106119, doi: 10.1016/j.knosys.2020.106119.
[42] C. Wu, F. Wu, T. Qi, and Y. Huang, "User modeling with click preference and reading satisfaction for news recommendation," in *Proc. 29th Int. Joint Conf. Artif. Intell.*, C. Bessiere, Ed., Jul. 2020, pp. 3023–3029, doi: 10.24963/IJCAI.2020/418.
[43] C. Wu, F. Wu, T. Qi, and Y. Huang, "FeedRec: News feed recommendation with various user feedbacks," *CoRR*, vol. abs/2102.04903, pp. 1–9, Feb. 2021.
[44] Y. Wu, C. DuBois, A. X. Zheng, and M. Ester, "Collaborative denoising auto-encoders for top-N recommender systems," in *Proc. 9th ACM Int. Conf. Web Search Data Mining*, Feb. 2016, pp. 153–162.
[45] Q. Yang, Y. Zhang, W. Dai, and S. J. Pan, *Transfer Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2020.
[46] Y. Zang and X. Hu, "LKT-FM: A novel rating pattern transfer model for improving non-overlapping cross-domain collaborative filtering," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, in Lecture Notes in Computer Science, vol. 10535, M. Ceci, J. Hollmén, L. Todorovski, C. Vens, and S. Dzeroski, Eds. Skopje, Macedonia: Springer, Sep. 2017, pp. 641–656, doi: 10.1007/978-3-319-71246-8_39.
[47] Q. Zhang, P. Hao, J. Lu, and G. Zhang, "Cross-domain recommendation with semantic correlation in tagging systems," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Budapest, Hungary, Jul. 2019, pp. 1–8, doi: 10.1109/IJCNN.2019.8852049.
[48] Q. Zhang, D. Wu, J. Lu, F. Liu, and G. Zhang, "A cross-domain recommender system with consistent information transfer," *Decis. Support Syst.*, vol. 104, pp. 49–63, Dec. 2017.
[49] Z. Zhang, X. Jin, L. Li, G. Ding, and Q. Yang, "Multi-domain active learning for recommendation," in *Proc. 30th AAAI Conf. Artif. Intell.*, D. Schuurmans and M. P. Wellman, Eds. Phoenix, AZ, USA: AAAI Press, 2016, pp. 2358–2364. [Online]. Available: http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12369
[50] L. Zhao, S. J. Pan, and Q. Yang, "A unified framework of active transfer learning for cross-system recommendation," *Artif. Intell.*, vol. 245, pp. 38–55, Apr. 2017, doi: 10.1016/j.artint.2016.12.004.
[51] Y. Zhen, W.-J. Li, and D.-Y. Yeung, "TagiCoFi: Tag informed collaborative filtering," in *Proc. 3rd ACM Conf. Recommender Syst. (RecSys)*. New York, NY, USA: Assoc. Comput. Machinery, 2009, pp. 69–76, doi: 10.1145/1639714.1639727.
[52] F. Zhu, Y. Wang, C. Chen, J. Zhou, L. Li, and G. Liu, "Cross-domain recommendation: Challenges, progress, and prospects," *CoRR*, vol. abs/2103.01696, pp. 1–10, Mar. 2021.

QINQIN WANG received the B.A. degree in software engineering from the Beijing University of Technology and the M.Sc. degree in information security from University College London. He is currently pursuing the Ph.D. degree with the Insight Centre for Data Analytics, School of Computer Science, University College Dublin, Ireland. His research interests include transfer learning, recommender systems, and machine learning.

DIARMUID OŔEILLY-MORGAN received the B.A. degree in English and film studies from Trinity College Dublin and the M.Sc. degree in computer science from University College Dublin (UCD). Since 2019, he has been working as a Research Assistant/an Engineer with the Insight Centre for Data Analytics, UCD. His research interests include the applicability of graph neural networks and federated learning to recommender systems.

ELIAS Z. TRAGOS received the master's degree in business administration (MBA) in techno-economics and the Ph.D. degree in wireless communications. He has been actively involved in many EU and national research projects as a Researcher, the Technical Manager, and a Project Coordinator. He has been the Technical Coordinator of the Smart Cities Project RERUM. He is currently a Research Project Manager with the Insight Centre for Data Analytics, UCD, Ireland. He has been extensively involved in the IERC, leading the Activity Chain 05 for Trusted IoT and co-leading the AC 03 for IoT Results Exploitation, contributing also every year in drafting the IoT strategic roadmap in the IERC Cluster Book. His research interests include the areas of machine learning, distributed artificial intelligence, wireless and mobile communications, the Internet of Things, cognitive radios, network architectures, fog computing, security, and privacy.

NEIL HURLEY received the M.Sc. degree in mathematical science from UCD, in 1988, and the Ph.D. degree with a focus on knowledge-based environments for engineering design, in 1995. In 1989, he joined the Hitachi Dublin Laboratory, a computer science research laboratory-based at Trinity College Dublin. A patent for the system was granted in the USA and Europe. His research focus switched to parallel and distributed computing and he became the Group Leader of the Parallel Computing Group, HDL, in 1995. During this period, he worked on the development of parallel simulation and optimization software, such as parallel finite element analysis and dynamic load balancing. In 1998, he was appointed as the Manager of HDL. He became a Lecturer with the UCD Computer Science Department, in 1999. He established the Information Hiding Laboratory, in 2001, a research laboratory focused on technology for embedding information in digital content. He has won over 1 million euro in research funding from Enterprise Ireland, Science Foundation Ireland, the EU, and industrial partners.

BARRY SMYTH received the Honrary Doctor of Technology (D.Tech.) degree (Hons.) from Robert Gordon University, U.K., in 2014. He holds the Digital Chair of Computer Science with University College Dublin. He is currently the Director of the Insight Centre for Data Analytics. He has been a fellow of the European Coordinating Committee on Artificial Intelligence (ECCAI), since 2003 and a member of the Royal Irish Academy, since 2011. He was the Director of the Clarity Centre for Sensor Web Technologies (2008–2013). He has previously held the position of the Head of the School for the School of Computer Science and Informatics, UCD. His research interests include the field artificial intelligence and include case-based reasoning, machine learning, recommender systems, user modeling, and personalization. Since 1992, he has been published over 400 peer-reviewed papers. His research has attracted more than 13,000 citations. He has an H-index of 58. He has received more than 20 best paper awards for his work. In 2014, he was named the SFI Researcher of the Year. His research interests extend beyond the laboratory and over years he has established a track-record for successfully translating his research into commercial opportunities and received numerous awards for his commercialization endeavor. In 2006, he was a Finalist of the Ernst and Young Entrepreneur of the Year and in 2012, he received the Irish Software Association's (ISA) Inaugural Award for Outstanding Academic Achievement. In 1999, he co-founded Changing Worlds based on his personalization research and helped to grow the company to more than 150 employees before it was acquired by Amdocs Inc., in 2008. In 2008, he co-founded HeyStaks based on his social search research. He has helped HeyStaks to raise more than 3.5 m in venture capital funding to date and continues to advise the company on its technology and market strategy. He is actively involved in the Irish startup scene as an Advisor and an Investor. He serves on the boards of a number of local startups. He is also a member of the Irish Times Trust.

AONGHUS LAWLOR received the Master of Science and Ph.D. degrees from University College Dublin (UCD). He is currently an Assistant Professor with the UCD and a Principal Investigator with the Insight Centre for Data Analytics. He has long experience in the areas of machine learning, artificial intelligence, explainable AI, natural language processing, and recommender systems. He has been actively involved in several EU and national research projects in the above mentioned areas and the Co-PI of very large industrial projects. He has also received several funded grants from Enterprise Ireland and the Science Foundation of Ireland, leading to patent applications.

RUIHAI DONG is currently an Assistant Professor with the School of Computer Science, University College Dublin. He has published in top peer-reviewed journals and conferences, such as WWW, RECSYS, IUI, and IJCAI, in 2018. His research interests include machine learning and deep learning, and their applications in recommender systems and finance. He was awarded the Outstanding Research Award 2018 by the UCD School of Computer Science for a series of significant publications that year. He has a track record of collaboration with industry and has worked with companies, including Eagle Alpha, SkillPages, and Samsung, and individually winning funding from Enterprise Ireland for commercialization studying of his research.