DaRE: A Cross-Domain Recommender System with Domain-aware Feature Extraction and Review Encoder

Yoonhyuk Choi
yoonhyuk95@snu.ac.kr
Seoul National University
Seoul, South Korea

Jiho Choi
jihochoi@snu.ac.kr
Seoul National University
Seoul, South Korea

Taewook Ko
taewook.ko@snu.ac.kr
Seoul National University
Seoul, South Korea

Chongkwon Kim*
ckim@snu.ac.kr
Seoul National University
Seoul, South Korea

ABSTRACT

Recent advents in recommender systems, especially in text-aided methods and Cross-Domain Recommendation (CDR), lead to promising results in solving data-sparsity and cold-start problems. Despite such progress, existing CDR approaches have some critical defects such as requiring overlapping users for the knowledge transfer or ignoring domain-aware features. In addition, text-aided methods, in general, emphasize aggregated item reviews and fail to capture the latent of individual reviews. To overcome such limitations, we propose a novel method, named Domain-aware Feature Extraction and Review Encoder (DaRE), which consists of the key components; domain-aware text analysis module, and review encoders. DaRE debilitates noises by separating domain-invariant features from domain-specific features through selective adversarial training. Then, with the features extracted from aggregated reviews, the review encoder fine-tunes the representations by aligning them with the features derived from individual reviews. The experiments on four real-world datasets show the superiority of DaRE over state-of-the-art single-domain and cross-domain methodologies, achieving 9.2 % and 3.6 % improvements, respectively.

CCS CONCEPTS

• **Information Systems** → **Recommender Systems**.

KEYWORDS

Cross-Domain Recommendation; Disentangled Representation Learning; Domain Adaptation; Textual Analysis; Data Sparsity

ACM Reference Format:
Yoonhyuk Choi, Jiho Choi, Taewook Ko, and Chongkwon Kim. 2021. DaRE: A Cross-Domain Recommender System with Domain-aware Feature Extraction and Review Encoder. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/

*Corresponding author

Conference'17, July 2017, Washington, DC, USA
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-XXXX-X/18/06…$15.00
https://doi.org/10.1145/

1 INTRODUCTION

With the rapid growth of e-commerce, recommender systems have become an obligatory tool for interconnecting customers with relevant items. Early schemes suffer from the cold-start problem caused by data insufficiency. To tackle the problems, some of them exploit auxiliary information such as social relations [12], the trustworthiness of reviewers [1], item images [40], and textual information. Especially, textual or linguistic information such as reviews are commonly available, and many text-aided recommendation algorithms have been introduced [6, 9, 10, 36, 45].

Most text-aided schemes deal with aggregated reviews rather than individual texts since aggregation provides richer information. Some studies infer the preferences of users by applying NLP techniques such as topic modeling [2, 28] to aggregated texts. More recently, [13, 44, 45] adopt DNN-based FEs (Feature Extractors), while others utilize attention mechanism [6, 10]. Extracted preferences or features are fed into prediction modules in the forms of MF or MLP. Contrary to the prior methods that ignore individual texts, we propose to utilize individual texts as well as aggregated reviews, simultaneously. We extract features via two different routes, one from aggregated reviews and the other from individual texts, and align them using a review network.

Along with the text-based recommender systems, numerous Cross-domain recommendation (CDR) algorithms [13, 18, 42] and transfer learning methods [16, 41] have been introduced. CDR leverages the information learned from source domains to improve a recommendation quality in a target domain. Some schemes [38, 41] use network fine-tuning, but they may suffer from catastrophic forgetting [7]. Context mapping techniques [15, 23, 25, 42], which map shareable information from a source to the target domain, is another branch of transfer learning popularly adopted in CDRs. These approaches require the same contexts like overlapped users or features, a restriction that can confine their applicability severely [19]. In real-world datasets, overlapped users are scarce and, more importantly, the contributions of overlapped users may not be significant, prompting further investigations for a model oblivious to overlapped users. As one solution, some CDR algorithms focus on capturing domain-invariant features common to both source and target domains [5, 14, 20, 30, 44], which is independent of sharing same contexts. Though domain common knowledge improve the accuracy of recommendations, italicizing this kind of information only may lead to sub-optimal performance, especially when two domains have different distributions.

To solve such constraints: *require duplicate users, only capturing domain-invariant features*, we propose a novel domain adaptation algorithm that extracts domain-aware knowledge from review texts, including both domain-invariant and domain-specific features without depending on the user or item overlap. Domain adversarial approaches extract domain-invariant features by minimizing source risk as well as H-divergence [4, 14, 21, 22]. However, these methods are highly dependent on the consistency of source and target distributions (*e.g., similar category*). To mitigate such restriction, we extend the concept of domain adaptation, which emphasizes the extraction of domain-specific feature to distill pertinent knowledge from multiple seemingly counterproductive domains.

Summarizing the above insights, Domain-aware Feature Extraction and Review Encoder (DaRE) adopts the following key components; domain-aware feature extraction, and a review encoder. Two mechanisms are closely coupled and interact with each other. DaRE extracts domain-aware features using three pairs of FEs, two for domain-specific features and one for domain-invariant feature. For domain-aware learning, we cleverly utilizes the domain adversarial technique that adequately controls the importance of domain-specific features or domain-invariant features. Through a selective adoption of the gradient reversal layer, three FEs are trained to capture these features. Another property of DaRE is that it utilizes individual texts as well as aggregated reviews. The review encoder network fine-tunes representations extracted from aggregated texts by aligning them with those extracted from individual reviews. The example of domain-aware feature extraction and review encoder can be seen in Figure 9 and 10, respectively.

We perform extensive experiments on real datasets to compare DaRE with several state-of-the-art algorithms. Experimental results show that DaRE outperforms all baselines. Also, the ablation studies scrutinize the contributions of the key modules of DaRE, showing all components are indispensable to performance enhancement.

The contributions of our work is summarized as follows.

- We propose a novel algorithm that adaptively extracts domain-invariant and domain-specific features depending on the characteristics of the source and target distributions. DaRE is a comprehensive method where domain-awareness is closely integrated with text-based feature extraction. Its domain adversarial gradient updates eventually modify the text-based FEs and capture domain-specific and domain-invariant features concurrently.
- Unlike previous CDR approaches that require duplicate entities such as common users or items from heterogeneous domains, DaRE focuses on retrieving review information that is domain-independent. Consequently, DaRE enjoys wide applicability and can be applied regardless of source and target domain homogeneity or heterogeneity.
- We propose a unique review encoder that compares features extracted in two routes. It fine-tunes features extracted from aggregated information using features from individual reviews. The review encoder improves the accuracy of representation and the quality of recommendation.
- We perform extensive experiments to answer the important research questions. Our results indicate the superiority of DaRE and the effectiveness of all key modules.

2 RELATED WORK

We delve into the two types of categories; text-based recommender algorithms and CDR.

2.1 Text-aided Recommender System

Textual information is the most popular side information and many text-based methods [8–10, 45] have been proposed. Most techniques integrate DNN based feature extraction with MF for a rating prediction. DeepCoNN [45] utilizes two parallel CNNs, while NPT [24] adopts Gated Recurrent Units for review analysis. Attention mechanisms are widely used also to pinpoint useful words and reviews [6, 10, 34, 35]. Even though prior works show the usefulness of textual information, the limitation of review information due to the limited size of training data, the irrelevance of reviews toward target items have been raised also [33, 43]. Unlike the prior works, DaRE utilizes both aggregated reviews and individual reviews. Also, DaRE couples the domain adversarial mechanism with three text-based FEs. To the best of our knowledge, this is the first attempt that integrates the two mechanisms.

2.2 CDR (Cross-Domain Recommendation)

CDR utilizes information from source domains to alleviate the cold-start problem in the target domain. Early studies [11, 26] adopt feature mapping technique that requires overlapped users. For example, RC-DFM [13] applies Stacked Denoising Autoencoder (SDAE) to each domain, where the learned knowledge of the same set of users are transferred from source to target domain. To overcome the restrictive requirement of overlapped users, CDLFM [37] and CATN [44] employ neighbor or similar user-based feature mapping. However, this kind of cross-domain algorithm implicates defects [46] like filtering noises or requiring duplicate users.

DA (Domain Adaptation) with the powerful mechanism of adversarial training has been adopted for various fields; VQA [32] for question answering, DAREC [42] for a recommendation. TDAR [39] assumes no user or item overlap and extracts domain-invariant textual features. However, these approaches focus on domain shareable knowledge ignoring domain-specific features.

MMT [20] considers domain-specific features as well as domain-invariant knowledge, but it simply adopts user and item embedding as domain-specific features. DADA [30] suggests domain-agnostic learning, but its domain discriminator focuses on domain-invariant feature extraction only also, leaving domain-specific feature extraction to be solely guided for the minimization of mutual information with MINE [3], which has proven to have some defects [27]. DaRE clears the aforementioned limitations, adopting a framework for the simultaneous extraction of domain-specific and domain-invariant knowledge through the modification of domain adaptation.

3 PROBLEM DEFINITION

Assume two datasets, D^s and D^t, be the information from the source and target domains, respectively. Each dataset consists of tuples, $(u, i, y_{u,i}, r_{u,i})$ which represents an individual review $r_{u,i}$ written by

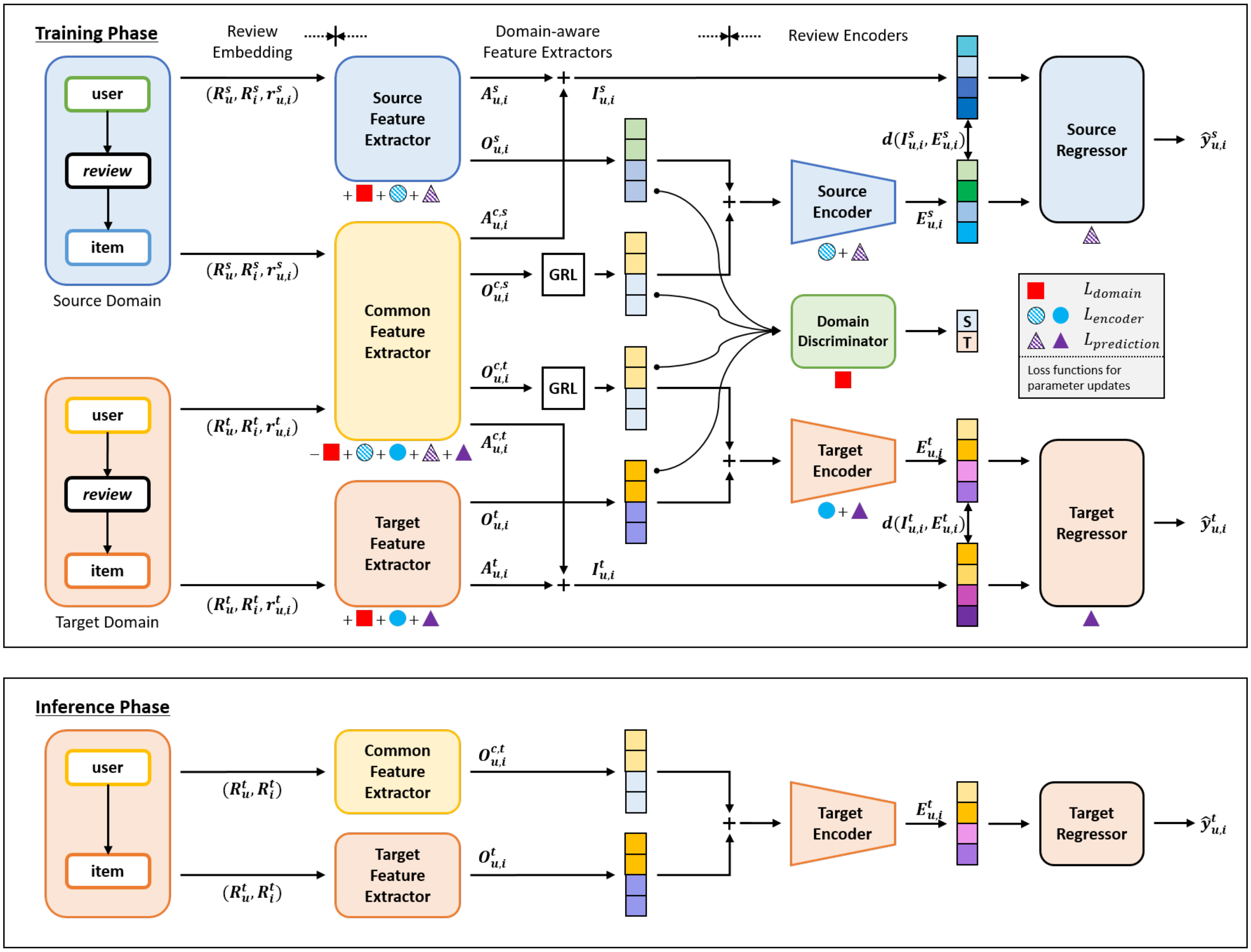


Figure 1: The overall architecture of DaRE for cross-domain recommendation

a user u for item i with a rating $y_{u,i}$. The two datasets take the form of $D^s = (u^s, i^s, y^s_{u,i}, r^s_{u,i})$ and $D^t = (u^t, i^t, y^t_{u,i}, r^t_{u,i})$, respectively. The goal of our task is to predict an accurate rating score $y^t_{u,i}$, using D^s and a partial set of D^t. A detailed explanation of the notations can be seen in Table 1.

3.1 Overview of DaRE

On the upper side of Figure 1 (training phase), our model Domain-aware Feature Extraction and Review Encoder (DaRE) starts with review embedding layers followed by three types of feature extractors (FEs). Integrated with domain discriminator, three FEs are trained independently for the parallel extraction of domain-specific O^s, O^t and domain-common knowledge $O^{c,s}, O^{c,t}$. Then, for each domain, the review encoder generates a single vector E^s, E^t with extracted features O by aligning them with individual review I^s, I^t. Finally, the regressor predicts an accurate rating that the user will give on an item. Here, shared parameters across two domains are common FE and a domain discriminator. We now explain each component precisely.

3.2 Review Embedding and Feature Extraction

We adopt text analysis method [45] that extracts user and item features from aggregated reviews using two parallel networks. Unlike the previous techniques [6, 10, 45] that use one or two pairs of parallel networks, DaRE adopts three pairs of FEs (Feature Extractors), FE^s, FE^c, and FE^t, named source, common, and target, for the separation of domain-specific, domain-common knowledge. The three FEs share the same architecture with unshared parameters, $\theta^s_{fe}, \theta^c_{fe}, \theta^t_{fe}$. As illustrated in Figure 2, each FE consists of a user feature extraction network FE^k_u and an item feature extraction network FE^k_i. To distinguish the domain identifier from the FE identifier, we use a superscript d to denote the domains of datasets (s for source and t for target) and k to represent three FE types (s, c, t). Note that the common FE (i.e. FE^c) uses both source and target domain reviews as an input.

Table 1: Definition of Notations

D^s, D^t	Source and target domain datasets
$u, i, y_{u,i}, r_{u,i}$	User, item, rating, and review
R_u, R_i	Aggregated reviews for user and item
$r_{u,i}$	Individual review for specific user and item
V_u, V_i	Embedding of aggregated reviews R_u, R_i
$A_{u,i}$	Embedding of individual review $r_{u,i}$
FE^k	Three types of feature extractors, $k = (s, c, t)$
F^k_u, F^k_i	User, item convolutional filter
$O_{u,i}$	Extracted feature of V_u, V_i
$\widehat{d}_{u,i}$	Predicted domain label of $O_{u,i}$
$I_{u,i}$	Extracted feature of $A_{u,i}$
$E_{u,i}$	Encoded feature of $O_{u,i}$
$\widehat{y}_{u,i}$	Predicted rating of $E_{u,i}$
GRL	Gradient Reversal Layer
θ	Model's parameters
$\oplus$	Concatenation operator
α, β	Weight hyper-parameters
μ	Learning ratio
$\mathcal{L}$	Loss function

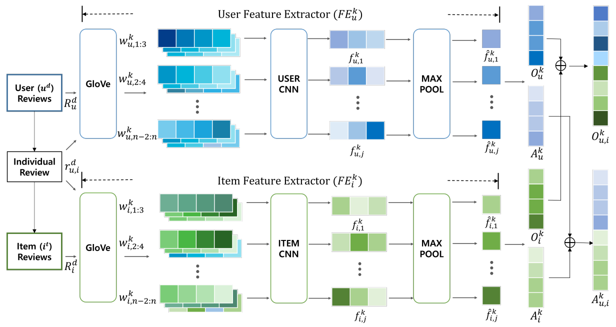
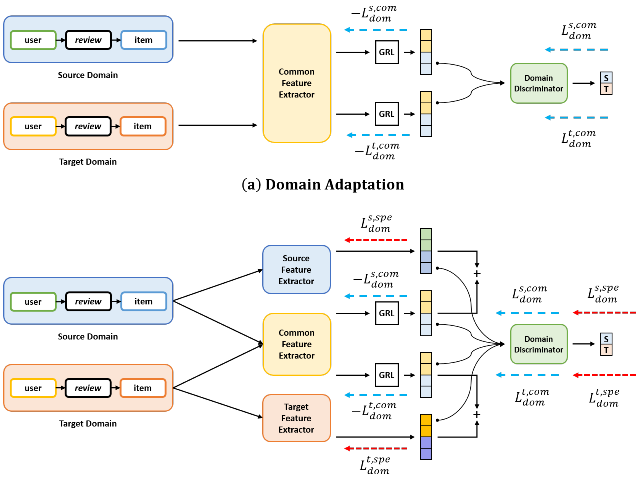


Figure 2: The architecture of a single review feature extractor. DaRE has three parallel review feature extractors of the same architecture with different inputs and parameters.

Another distinctive characteristic is that DaRE utilizes individual texts as well as aggregated reviews, simultaneously. First, all reviews (except for individual review $r^d_{u,i}$) written by user u^d are concatenated to a single R^d_u. Likewise, R^d_i for each item i^d is the concatenation of user reviews. Each individual review, $r^d_{u,i}$, fine-tunes the final representations of R^d_u and R^d_i through the review encoder. An FE begins with a word embedding layer. We utilize first n *words* in R^d_u or R^d_i and adopt GloVe [31] for word vectorization. The words are mapped to c-dimensional vectors $\phi(w_n)$. Word vectors are concatenated to form $V^d_u = \phi(w_1) \oplus \phi(w_2) \oplus ... \oplus \phi(w_n)$, where ϕ and $\oplus$ are embedding and concatenation function, respectively. Then 2-D convolutional network with filters $F \in \mathbb{R}^{h \times c}$ extract features from V^d_u, followed by non-linear activation function ReLU with max pooling layer. Specifically, j-th filter, F^s_j for user feature extraction yields $f^k_{u,j}$ as follows:

$$f^k_{u,j} = max(ReLU(V^d_{u,1:n} \cdot F^k_{u,j} + b_j)) \tag{1}$$



(a) **Domain Adaptation**

(b) **Domain-aware Feature Extraction**

Figure 3: The architecture of (a) Domain adaptation, and (b) Domain-aware feature extraction. The dotted line (blue and red) denotes back-propagation for domain loss

, b_j is a bias term. The representation O^k_u, O^k_i of user u^d and item i^d is the concatenation of the scalar outputs $f^k_{u,j}$,

$$\begin{aligned} O^k_u &= [f^k_{u,1}, f^k_{u,2}, \cdots, f^k_{u,j}, \cdots, f^k_{u,J}], \\ O^k_i &= [f^k_{i,1}, f^k_{i,2}, \cdots, f^k_{i,j}, \cdots, f^k_{i,J}] \end{aligned} \tag{2}$$

, where J is the number of filters. The final representation $O^k_{u,i}$ is a simple concatenation of O^k_u and O^k_i.

$$O^k_{u,i} = [O^k_u \oplus O^k_i] \tag{3}$$

Likewise, we derive the embedding $A^k_{u,i}$ of an individual review $r^d_{u,i}$ with three FEs as follows:

$$\begin{aligned} A^k_u = FE^k_u(r^d_{u,i}),\ A^k_i = FE^k_i(r^d_{u,i}) \\ A^k_{u,i} = [A^k_u \oplus A^k_i] \end{aligned} \tag{4}$$

As shown in Figure 1, the input for source and target FEs are $(R^s_u, R^s_i, r^s_{u,i})$ and $(R^t_u, R^t_i, r^t_{u,i})$, respectively. The common FE accepts both inputs. The outputs of three FEs are denoted as follows: $A^s_{u,i}, O^s_{u,i}$ from FE^s and $A^t_{u,i}, O^t_{u,i}$ from FE^t and $A^{c,s}_{u,i}, O^{c,s}_{u,i}, A^{c,t}_{u,i}, O^{c,t}_{u,i}$ from FE^c.

3.3 Integrating Domain Discriminator for Domain-aware Feature Extraction

We propose a mechanism that separates the domain-invariant features from the domain-specific features. The DANN [14] architecture, which utilizes a domain-shareable module, effectively transfers knowledge between two different domains. As can be seen in Figure 3-(a), a domain discriminator gives a penalty to prevent a common FE from capturing domain-specific knowledge.

One drawback of DANN is that it is vulnerable to domain-mismatches (e.g., categories) resulted in prohibitive applicability. To subjugate the limitation, we additionally adopt source and target FEs to capture domain-specific knowledge. Figure 3-(b) is the architecture of the proposed scheme. The domain-awareness module is closely coupled with the three text-based FEs explained before. Using the domain of reviews as a label, the domain discriminator prevents a common FE from capturing domain-specific features. On the contrary, the domain discriminator penalizes the source and the target FEs if source feature $O^s_{u,i}$ or target feature $O^t_{u,i}$ has insufficient domain-specific information. In this way, our scheme can adaptively secure robustness against similarity or difference between source and target domains; if source and target domains are different, then domain-specific features will be emphasized, and vice versa.

For a domain discriminator, we utilize two layers of a fully-connected neural network as below:

$$\widehat{d}^k_{u,i} = H_2 g(H_1 O^k_{u,i} + b_1) + b_2 \tag{5}$$

, where g is an activation function and H_1, H_2 are the parameters of the MLP. $\widehat{d}^k_{u,i}$ denotes predicted domain label of feature $O^k_{u,i}$. A domain loss can be calculated through binary cross-entropy between true $d^k_{u,i}$ and predicted $\widehat{d}^k_{u,i}$ label as follows:

$$\begin{gathered}
\mathcal{L}^{s,com}_{dom}, \mathcal{L}^{t,com}_{dom} = -\frac{1}{N_s}\sum_{u,i\in D^s} log(1-\widehat{d}^{c,s}_{u,i}), -\frac{1}{N_t}\sum_{u,i\in D^t} log(\widehat{d}^{c,t}_{u,i}) \\
\mathcal{L}^{s,spe}_{dom}, \mathcal{L}^{t,spe}_{dom} = -\frac{1}{N_s}\sum_{u,i\in D^s} log(1-\widehat{d}^{s}_{u,i}), -\frac{1}{N_t}\sum_{u,i\in D^t} log(\widehat{d}^{t}_{u,i}) \\
\mathcal{L}_{dom} = \mathcal{L}^{s,com}_{dom} + \mathcal{L}^{t,com}_{dom} + \mathcal{L}^{s,spe}_{dom} + \mathcal{L}^{t,spe}_{dom},
\end{gathered} \tag{6}$$

where N_s and N_t are the number of training data in source and target domains, respectively. The domain label $d^k_{u,i}$ is a binary value, $\{0, 1\}$ for source and target, respectively. The proof of Equation 6 can be seen in supplementary material.

Through Figure 3-(b), we can see that a domain discriminator is being updated with four types of losses. Then, two losses $-\mathcal{L}^{s,com}_{dom}, -\mathcal{L}^{t,com}_{dom}$ (blue arrows) with Gradient Reversal Layer (GRL) updates the common FE, while another losses $\mathcal{L}^{s,spe}_{dom}, \mathcal{L}^{t,spe}_{dom}$ (red arrows) updates source, target FE, respectively. During back-propagation, GRL multiplies a negative constant, which is positioned between common FE and domain discriminator. Thus, the common FE is trained to capture *domain-indiscriminative* knowledge: fooling domain discriminator. This reinforces common convolution filter F^c. On the contrary, domain loss without GRL prompts the source and the target FEs to capture *domain-discriminative* information: minimizing domain loss, which updates source or target convolution filters F^s, F^t.

3.3.1 ***Remark.*** The domain-aware feature extraction employs the idea of domain disentanglement [30] but they differ in two perspectives. First, we selectively adopt GRL for domain-aware feature extraction without calculating mutual information. Second, our method notably focuses on the review analysis rather than image classification.

3.4 Review Encoder and Prediction

Figure 4 illustrates the architecture of a review encoder taking a user u_1 and an item i_4 as an example. Prior methods only utilize the gross review information such as R_{u_1} and R_{i_4} and ignore peculiar information $r_{1,4}$ that u_1 gives on i_4. Intuitively, to retrieve the detailed latent representations, we propose a novel review encoder that handles specific review $r_{u,i}$ from O_u and O_i.

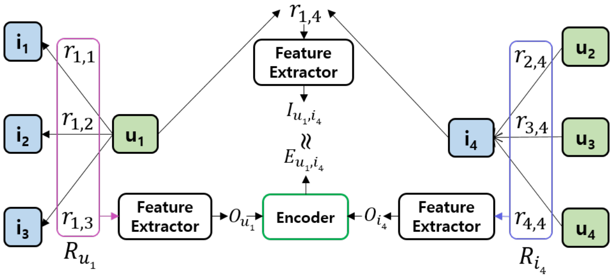


Figure 4: A simple architecture of review encoder

First, from an individual review, its final representation, $I^d_{u,i}$, is extracted as the summation of two vectors as shown in Figure 1:

$$I^d_{u,i} = A^{c,d}_{u,i} + A^d_{u,i} \tag{7}$$

Also, we can obtain the representation of user and item from aggregated texts. From aggregated reviews R^d_u, R^d_i, FEs obtain representations $O^{c,d}_{u,i}$ and $O^d_{u,i}$, respectively. We devise the review encoder to align the latent from an individual review $I^d_{u,i}$ with the latent from aggregated reviews. Two embeddings $O^{c,d}_{u,i}, O^d_{u,i}$ are loaded to the encoder (fully connected network), generating a final representation $E^d_{u,i}$ as:

$$E^d_{u,i} = Encoder^d(O^{c,d}_{u,i} + O^d_{u,i}) \tag{8}$$

To align the two vectors, $I^d_{u,i}, E^d_{u,i}$, we adopt Euclidean distance as the loss function,

$$\mathcal{L}^d_{enc} = d(E^d_{u,i}, I^d_{u,i}) = ||E^d_{u,i} - I^d_{u,i}||^2_2 \tag{9}$$

The regressor predicts a rating score that user u^d will give to the item i^d. A single deep feed-forward neural network consists of two layers serves as the regressor. For each domain, the predicted ratings $E^d_{u,i}$ and $I^d_{u,i}$ are,

$$\begin{aligned}
\widehat{y}^{d,O}_{u,i} &= Regressor^d(E^d_{u,i}) = W^d_2 g(W^d_1 E^d_{u,i} + b^d_1) + b^d_2 \\
\widehat{y}^{d,I}_{u,i} &= Regressor^d(I^d_{u,i}) = W^d_2 g(W^d_1 I^d_{u,i} + b^d_1) + b^d_2,
\end{aligned} \tag{10}$$

where $\widehat{y}^{d,O}_{u,i}$ is a predicted rating based on aggregated reviews and $\widehat{y}^{d,I}_{u,i}$ is an inferred rating from an individual review in domain d. Finally, we can define a regression loss function: the difference between predicted scores $\widehat{y}^{d,O}_{u,i}, \widehat{y}^{d,I}_{u,i}$ and true label $y^d_{u,i}$. We adopt MSE loss for the objective function as below:

$$\mathcal{L}^d_{reg} = \frac{1}{N_d}\sum_{u,i\in D_d}\left((\widehat{y}^{d,O}_{u,i} - y^d_{u,i})^2 + (\widehat{y}^{d,I}_{u,i} - y^d_{u,i})^2\right) \tag{11}$$

3.5 Inference Phase

During the inference phase, DaRE utilizes the trained modules (common and target FEs, target review encoder, target regressor) with the entire aggregated reviews of user and item. Finally, the rating prediction in a target domain is as follows:

$$\begin{aligned} O^{c,t}_{u,i}, O^{t}_{u,i} &= [FE^{c}_{u}(R^{t}_{u}) \oplus FE^{c}_{i}(R^{t}_{i})], [FE^{t}_{u}(R^{t}_{u}) \oplus FE^{t}_{i}(R^{t}_{i})], \\ E^{t}_{u,i} &= Encoder^{t}(O^{c,t}_{u,i} + O^{t}_{u,i}), \\ \hat{y}^{t,O}_{u,i} &= Regressor^{t}(E^{t}_{u,i}) \end{aligned} \tag{12}$$

3.6 Optimization Strategy

For the optimization, we can define the objective function through the weighted sum of three Equations 6, 9, 11 as below:

$$\mathcal{L}^{d}_{total} = \alpha\mathcal{L}_{dom} + \beta\mathcal{L}^{d}_{enc} + \mathcal{L}^{d}_{reg} + \gamma||\theta|| \tag{13}$$

The hyper-parameters α, β balance the domain and encoder losses. From our experiment, $\alpha = 0.1$, $\beta = 0.05$ yield the best performance (details are in supplementary material). θ denotes all parameters of our model and γ is a regularization term. We update the model's parameters through the gradient descent by minimizing Equation 13, where the parameter update for each module can be seen in Figure 1 with three basic shapes. A shape with a horizontal stripe denotes a loss from the source domain. For training, we adopt a mini-batch with Adam optimizer $lr = 1e^{-4}$ and early stopping. The trainable parameters of DaRE are three FEs, a domain discriminator, two review encoders, and two regressors. We now explain the parameter updates of each module in detail.

First, let us assume the parameters of source and target regressor as $\theta^{s}_{reg}, \theta^{t}_{reg}$. The loss function associated with the update of θ^{d}_{reg}, is a regression error defined in Equation 11 (see purple triangles in Figure 1). We can simply update the parameters of regression layer, θ^{s}_{reg} and θ^{t}_{reg} with proper learning rate μ as follows:

$$\theta^{s*}_{reg} = \theta^{s}_{reg} - \mu\frac{\partial\mathcal{L}^{s}_{pred}}{\partial\theta^{s}_{reg}}, \quad \theta^{t*}_{reg} = \theta^{t}_{reg} - \mu\frac{\partial\mathcal{L}^{t}_{pred}}{\partial\theta^{t}_{reg}} \tag{14}$$

Then, we can define the losses for the update of two review encoders θ^{d}_{enc} using chain rule:

$$\begin{aligned} \theta^{s*}_{enc} &= \theta^{s}_{enc} - \mu(\beta\frac{\partial\mathcal{L}^{s}_{enc}}{\partial\theta^{s}_{enc}} + \frac{\partial\mathcal{L}^{s}_{pred}}{\partial\theta^{s}_{reg}}\frac{\partial\theta^{s}_{reg}}{\partial\theta^{s}_{enc}}), \\ \theta^{t*}_{enc} &= \theta^{t}_{enc} - \mu(\beta\frac{\partial\mathcal{L}^{t}_{enc}}{\partial\theta^{t}_{enc}} + \frac{\partial\mathcal{L}^{t}_{pred}}{\partial\theta^{t}_{reg}}\frac{\partial\theta^{t}_{reg}}{\partial\theta^{t}_{enc}}) \end{aligned} \tag{15}$$

, where $\mathcal{L}^{d}_{enc} = ||E^{d}_{u,i} - I^{d}_{u,i}||^{2}_{2}$ (blue circles in Figure 1).

For a domain discriminator, the parameter θ_{dom} is updated with the summation of domain-invariant $\mathcal{L}^{s,com}_{dom}, \mathcal{L}^{t,com}_{dom}$ and domain-specific $\mathcal{L}^{s,spe}_{dom}, \mathcal{L}^{t,spe}_{dom}$ losses (red squares in Figure 1), which is defined in Equation 6:

$$\theta^{*}_{dom} = \theta_{dom} - \mu\frac{\partial\mathcal{L}_{dom}}{\partial\theta_{dom}} \tag{16}$$

Finally, we can define the update functions for three FEs $\theta^{s}_{fe}, \theta^{c}_{fe}, \theta^{t}_{fe}$, integrating three types of losses as follows:

$$\begin{aligned} \theta^{s*}_{fe} &= \theta^{s}_{fe} - \mu\Bigg\{\alpha\frac{\partial(\mathcal{L}^{s,com}_{dom} + \mathcal{L}^{s,spe}_{dom})}{\partial\theta_{dom}}\frac{\partial\theta_{dom}}{\partial\theta^{s}_{fe}} + \\ &\beta(\frac{\partial\mathcal{L}^{s}_{enc}}{\partial\theta^{s}_{enc}}\frac{\partial\theta^{s}_{enc}}{\partial\theta^{s}_{fe}} + \frac{\partial\mathcal{L}^{s}_{pred}}{\partial\theta^{s}_{reg}}\frac{\partial\theta^{s}_{reg}}{\partial\theta^{s}_{enc}}\frac{\partial\theta^{s}_{enc}}{\partial\theta^{s}_{fe}})\Bigg\}, \\ \theta^{c*}_{fe} &= \theta^{c}_{fe} - \mu\Bigg\{\alpha\frac{\partial(\mathcal{L}^{s,spe}_{dom} - \mathcal{L}^{s,com}_{dom} + \mathcal{L}^{t,spe}_{dom} - \mathcal{L}^{t,com}_{dom})}{\partial\theta_{dom}}\frac{\partial\theta_{dom}}{\partial\theta^{c}_{fe}} \\ &+ \beta\left(\frac{\mathcal{L}^{s}_{enc}}{\partial\theta^{s}_{enc}}\frac{\partial\theta^{s}_{enc}}{\partial\theta^{c}_{fe}} + \frac{\mathcal{L}^{t}_{enc}}{\partial\theta^{t}_{enc}}\frac{\partial\theta^{t}_{enc}}{\partial\theta^{c}_{fe}}\right) \\ &+ \frac{\partial\mathcal{L}^{s}_{pred}}{\partial\theta^{s}_{reg}}\frac{\partial\theta^{s}_{reg}}{\partial\theta^{s}_{enc}}\frac{\partial\theta^{s}_{enc}}{\partial\theta^{c}_{fe}} + \frac{\partial\mathcal{L}^{t}_{pred}}{\partial\theta^{t}_{reg}}\frac{\partial\theta^{t}_{reg}}{\partial\theta^{t}_{enc}}\frac{\partial\theta^{t}_{enc}}{\partial\theta^{c}_{fe}}\Bigg\}, \\ \theta^{t*}_{fe} &= \theta^{t}_{fe} - \mu\Bigg\{\alpha\frac{\partial(\mathcal{L}^{t,com}_{dom} + \mathcal{L}^{t,spe}_{dom})}{\partial\theta_{dom}}\frac{\partial\theta_{dom}}{\partial\theta^{t}_{fe}} + \\ &\beta(\frac{\partial\mathcal{L}^{t}_{enc}}{\partial\theta^{s}_{enc}}\frac{\partial\theta^{t}_{enc}}{\partial\theta^{t}_{fe}} + \frac{\partial\mathcal{L}^{t}_{pred}}{\partial\theta^{t}_{reg}}\frac{\partial\theta^{t}_{reg}}{\partial\theta^{t}_{enc}}\frac{\partial\theta^{t}_{enc}}{\partial\theta^{t}_{fe}})\Bigg\} \end{aligned} \tag{17}$$

4 EXPERIMENTS

In this section, we conduct experiments with multiple real-world datasets of different categories to answer the following research questions:

- **RQ1:** Does DaRE outperforms compared with several state-of-the-art recommendation approaches?
- **RQ2:** Does the core components of DaRE: domain-aware feature extraction and review encoder, are essential for the recommendation quality?
- **RQ3:** How well the properties of domain-aware knowledge and peculiar information are preserved in extracted features?

4.1 Experiment Setup

We systematically conduct experiments with publicly available datasets *Amazon*[1] and *Yelp*[2]. The target domain includes the following four categories of Amazon: Office Products(OP), Automotive(Au), Patio Lawn and Garden(PL), and Instant Video(IV). The source domain consists of four datasets: three categories, Baby, Kindle Store(KS), Toys and Games(TG) from *Amazon* and one from *Yelp*. We adopt *Yelp* data to scrutinize the effect of excluding duplicate users for CDR scenario. Also, to lens on the cold-start problem, we designate datasets with sparse interactions as the target domain. The statistical details of datasets are summarized in Table 2.

Like previous studies, each target dataset is divided into three parts: 80 percent for training, 10 percent for validation, and another 10 percent for testing. We randomly sample source domain data such that its size equals that of the target domain data. For word embedding, we use Glove with a fixed embedding dimension of 100. We apply 100 convolution filters of size $F \in \mathbb{R}^{5\times100}$. The performance is

[1] http://jmcauley.ucsd.edu/data/amazon/
[2] https://www.yelp.com/dataset

Table 2: Statistics of the datasets

		# users	# items	# reviews	sparsity
Source	Baby	19,445	7,050	160,792	99.88%
	Kindle Store (KS)	68,223	61,934	982,619	99.97%
	Toys and Games (TG)	19,412	11,924	167,597	99.92%
	Yelp	1.9 M	0.2 M	8.1 M	99.99%
Target	Office Products (OP)	4,905	2,420	53,258	99.55%
	Automotive (Au)	2,928	1,835	20,473	99.62%
	Patio Lawn and Garden (PG)	1,686	962	13,272	99.18%
	Instant Video (IV)	5,130	1,685	37,126	99.57%

evaluated based on the validation score with early stopping under 300 iterations. We upload our *code*[3] for a reproducibility.

4.2 Baselines

We select 10 exemplary state-of-the-art single-domain and cross-domain approaches. Detailed explanations are as follows:

4.2.1 *Single-Domain Approaches*.

- **PMF** [29] is a classical probabilistic matrix factorization method.
- **NeuMF** [17] combines deep neural networks with a probabilistic model. NeuMF achieves great performance enhancement over classical methods.
- **DeepCoNN** [45] leverages review texts for rating prediction. They retrieve users' interest from textual reviews and jointly encode the latent of user and item with two parallel neural networks.
- **NARRE** [6] improves the *DeepCoNN* by employing attention mechanisms to measure the usefulness of reviews.
- **AHN** [10] proposes a hierarchical attention mechanism: review embedding is generated by applying sentence-level attention, followed by review-level attention for retrieving user and item embedding.

4.2.2 *Cross-Domain Approaches*.

- **DANN** [14] proposes the seminal domain adversarial technique that extracts domain-invariant features from two different domains. Here, review texts are embedded as 5,000-dimensional feature vectors.
- **DAREC** [42] assume the same set of users between two domains and integrate AutoEncoder with domain adaptation to transfer rating patterns from a source to the target domain.
- **RC-DFM** [13] fuses review texts with rating information. With SDAE, it preserves the latent features with rich semantic information. For a fair comparison, we additionally train the text convolution layer for each domain.
- **CATN** [44] transfers knowledge at an aspect level. The model extracts multiple aspects from review texts and learns aspect correlation across domains with an attention mechanism.
- **MMT** [20] adopt domain-invariant components shared across two domains. Here, we adopt text convolution layers for a knowledge transfer (review texts can act as domain-invariant information). Also, the trained parameter of the text convolution layer is retrained in a target domain for performance enhancement.

4.3 Performance Analysis (RQ1)

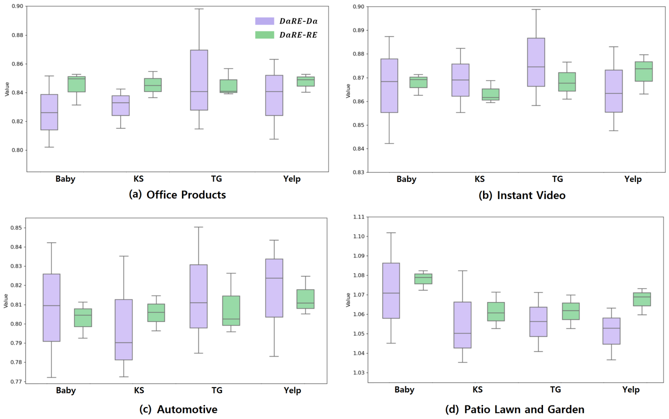
DaRE consistently outperforms all single-domain approaches. Table 3 shows the MSE score of DaRE with state-of-the-art approaches. For single-domain methodologies, rating-based PMF and NeuMF performed worse than text-based methods indicating the usefulness of textual information. NARRE and AHN which adopt attentions outperform DeepCoNN. Nonetheless, with the aid of domain-aware knowledge transfer and review encoder, DaRE outperforms AHN, which is competitive rather than attention mechanism.

Table 3: Comparison with Baselines, MSE Score

Target Domain Method & Source Domain		OP	IV	Au	PL
PMF		1.0852	1.1296	1.1617	1.1772
NeuMF		0.9742	1.0135	1.0871	1.1427
DeepCoNN		0.9018	0.9495	0.9789	1.1285
NARRE		0.8631	0.9139	0.8882	1.1076
AHN		0.8596	0.8922	0.8625	1.0939
DANN	Baby	0.9661	0.9855	0.9463	1.2290
	KS	0.9390	0.9461	0.8802	1.1896
	TG	0.9434	0.9872	0.9449	1.1996
	Yelp	1.1182	1.1472	1.1827	1.3946
DAREC	Baby	0.9889	1.0604	1.0002	1.1229
	KS	0.9875	1.0451	0.9972	1.1512
	TG	0.9724	1.0429	0.9927	1.1314
	Yelp	0.9940	1.0725	1.0042	1.1498
RC-DFM	Baby	0.8335	0.8783	0.7918	1.0940
	KS	0.8387	0.8545	0.8002	1.0958
	TG	0.8284	0.8678	0.7938	1.1087
	Yelp	0.8412	0.8716	0.8019	1.1120
CATN	Baby	0.8750	0.9153	0.8244	1.1408
	KS	0.8719	0.9064	0.8307	1.1438
	TG	0.8732	0.8923	0.8259	1.1292
	Yelp	0.8761	0.9193	0.8369	1.1493
MMT	Baby	0.8149	0.8619	0.8184	1.1155
	KS	0.8203	0.8552	0.7995	1.0993
	TG	0.8219	0.8777	0.8001	1.0942
	Yelp	0.8564	0.8811	0.8326	1.1168
DaRE	Baby	**0.7894**	0.8515	0.7853	**1.0281**
	KS	0.8147	**0.8330**	0.7977	1.0288
	TG	0.8101	0.8550	**0.7698**	1.0394
	Yelp	0.8064	0.8471	0.7842	1.0331
Improvement (%)		+ 3.2 %	+ 2.5 %	+ 2.8 %	+ 6.0 %

The quality of cross-domain recommendation can be degraded with respect to the domain discrepancy. We first investigate the performance of five cross-domain algorithms. Like DaRE, DANN and MMT transfer knowledge without user overlapping. The results show that DANN and MMT are unstable depending on

[3] https://anonymous.4open.science/r/DaRE-9CC9/



(a) Office Products

(b) Instant Video

(c) Automotive

(d) Patio Lawn and Garden

Figure 5: Performance of DaRE excluding core components; domain-aware feature extractor and review encoder

the pairing of source and target, which reveals the limitation of DANN and MMT. Typically, in the case of *Patio Lawn and Garden*, AHN with single-domain achieves the best performance among all baselines. This result demonstrates that utilizing additional domains without considering noises can degrade the recommendation quality. In contrast, DaRE shows stable performance highlighting that its domain-aware feature extraction effectively alleviates noises from the source domain.

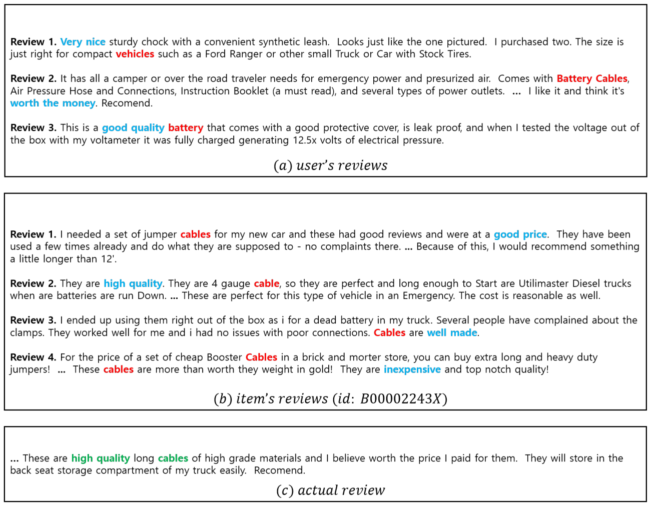
Knowledge transfer of overlapping users has limited contribution for the cross-domain recommendation quality. Some argued [19] that the knowledge transfer based on overlapping users has a limited impact on the overall performance. Note that DAREC, RC-DFM require duplicate users for a knowledge transfer. Specifically, DAREC, which utilizes rating information of duplicate users only, shows relatively low performance, again suggesting the usefulness of review information. Compared to DAREC, RC-DFM achieves the best performance for the two datasets. However, even excluding a knowledge transfer by removing duplicate users (selecting $Yelp$ as a source domain), the performance of DAREC and RC-DFM varies insignificantly (no more than 1% on average). Though CATN utilizes auxiliary reviews of another user who gave the same rating, does not show outstanding performance in our experiments. Instead, DaRE efficiently utilize additional domain through the adoption of review texts, which is less restrictive for the selection of a domain.

Summary. The performance of DaRE exceeds the state-of-the-art single (AHN) and cross-domain (RC-DFM or MMT) approaches about 9.2 % and 3.6 %, respectively.

4.4 Core Component Analysis (RQ2)

We assume two variants of DaRE to show the influence of excluding a domain-aware feature extractor and the review encoder. In Figure 5, the mean and variance of the variants of DaRE are plotted, with respect to each target domain. Here, the purple and green box denotes the performance of excluding domain-aware feature extractor (*DaRE - Da*), and review encoder (*DaRE - RE*), respectively.

Domain-aware feature extraction is fundamental for debilitating noises. First, we exclude domain discriminator from DaRE (*DaRE - Da*). Through Figure 5, we can see that the performance of *DaRE - Da* shows higher variance (purple boxes) compared to *DaRE - RE* (green boxes), suggesting the domain discrepancy is critical for overall performance. For example, in Figure 5-(b), the adoption of *Yelp* as source domain leads to the performance improvement for *Instant Video (IV)*, while the selection of *Toys and Games (TG)* can degrade the recommendation quality.



Review 1. Very nice sturdy chock with a convenient synthetic leash. Looks just like the one pictured. I purchased two. The size is just right for compact vehicles such as a Ford Ranger or other small Truck or Car with Stock Tires.

Review 2. It has all a camper or over the road traveler needs for emergency power and presurized air. Comes with Battery Cables, Air Pressure Hose and Connections, Instruction Booklet (a must read), and several types of power outlets. ... I like it and think it's worth the money. Recomend.

Review 3. This is a good quality battery that comes with a good protective cover, is leak proof, and when I tested the voltage out of the box with my voltameter it was fully charged generating 12.5x volts of electrical pressure.

(a) user's reviews

Review 1. I needed a set of jumper cables for my new car and these had good reviews and were at a good price. They have been used a few times already and do what they are supposed to - no complaints there. ... Because of this, I would recommend something a little longer than 12'.

Review 2. They are high quality. They are 4 gauge cable, so they are perfect and long enough to Start are Utilimaster Diesel trucks when are batteries are run Down. ... These are perfect for this type of vehicle in an Emergency. The cost is reasonable as well.

Review 3. I ended up using them right out of the box as i for a dead battery in my truck. Several people have complained about the clamps. They worked well for me and i had no issues with poor connections. Cables are well made.

Review 4. For the price of a set of cheap Booster Cables in a brick and morter store, you can buy extra long and heavy duty jumpers! ... These cables are more than worth they weight in gold! They are inexpensive and top notch quality!

(b) item's reviews (id: B00002243X)

... These are high quality long cables of high grade materials and I believe worth the price I paid for them. They will store in the back seat storage compartment of my truck easily. Recomend.

(c) actual review

Figure 6: With the output of FEs and review encoder, we highlight most similar words of user, item's past reviews

Review encoder contributes to the overall performance significantly. Compared to *DaRE - Da*, *DaRE - RE* shows relatively stable results independent of a category of source domain. Though the mechanism of domain-aware feature extraction effectively controls the mismatches between two different domains, we notice that the excluding review encoders are critical for the recommendation quality. Specifically, we notice that *DaRE - RE* shows relatively lower performance compared to *DaRE - Da* except for Figure 5-(c).

Here, we empirically show the significance of utilizing a domain-aware feature extractor, and the review encoder, respectively. Based on the above results, we contemplate that DaRE systematically improves the recommendation quality under the cross-domain scenario, through the integration of two novel mechanisms.

4.5 Explainability of DaRE (RQ3)

To provide intuitive analysis, we investigate the interpretability of DaRE. We adopt *Toys and Games* and *Automotive* as source and target domains, respectively. Figure 6-(a) and 6-(b) denote specific user and item reviews in the target domain. 6-(c) is a review that the user has written after purchasing the item. A detailed aspect of this item can be available in *Amazon Automotive* with its code (*id: B00002243X*).

First, we apply the common and target feature extractors for each review and retrieve its embedding vectors $O^{c,t}_{u,i}$, and $O^{t}_{u,i}$, respectively. Then, for each word embedding in Figure 6-(a) and (b), we highlight the most similar words, *e.g., cosine similarity*, compared to $O^{c,t}_{u,i}$ and $O^{t}_{u,i}$. Specifically, the blue highlighted words are the most similar to the output of the common feature extractor $O^{c,t}_{u,i}$, while the red highlighted words are the most similar to the output of the target feature extractor $O^{t}_{u,i}$. As can be seen, the blue words are more related to the semantic meaning of words which can be domain-indiscriminative (*e.g., nice and good quality*), while red words are relevant to domain-specific knowledge (*e.g., cable and vehicle*).

Similarly, in Figure 6-(c), we highlight top-2 similar words (green) compared to the output of review encoder $E^{t}_{u,i}$, which is generated from $O^{c,t}_{u,i}$ and $O^{t}_{u,i}$. Here, a review encoder well captures (*e.g., high quality and cable*) that the user will be interested in. To summarize, DaRE not only well predicts the accurate rating (4.89 for 5.0) but also demonstrates the vitality of domain-aware feature extractor and review encoder for capturing the transferable knowledge and relatedness between the user and item.

5 CONCLUSION

In this paper, we propose DaRE, a novel domain adaptation method utilizing review texts from multiple domains for a knowledge transfer. Compared to previous approaches, our method is able to capture domain-invariant and domain-specific information of different categories with the aid of domain-aware feature extraction. Moreover, we suggest the use of a review encoder, to better represent the attribute of reviews that the users will generate after purchasing a specific item. Extensive experiments and ablation studies on real datasets confirm the superiority of our method which is independent of users and items of different domains.

SUPPLEMENTARY MATERIAL

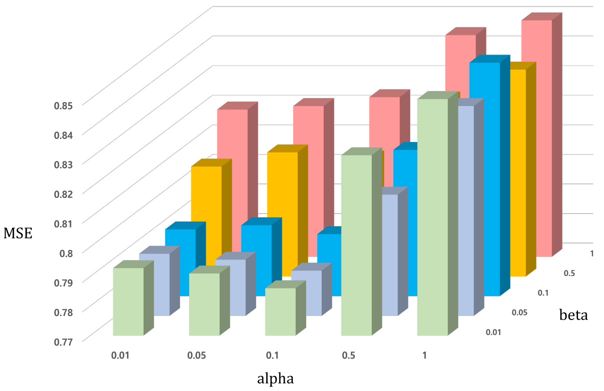


Figure 7: MSE score for varying parameter α, β

Hyper-parameter Analysis

In Equation 13, we define the two hyper-parameters α and β to balance the significance of domain and encoder loss. In Figure 7, the x-axis denotes the value of α, the y-axis is the value of β, and the z-axis demonstrates MSE score. For a simplified analysis, we simply visualize the performance of a single target domain *Instant Video*, adopting *Baby* as a source domain.

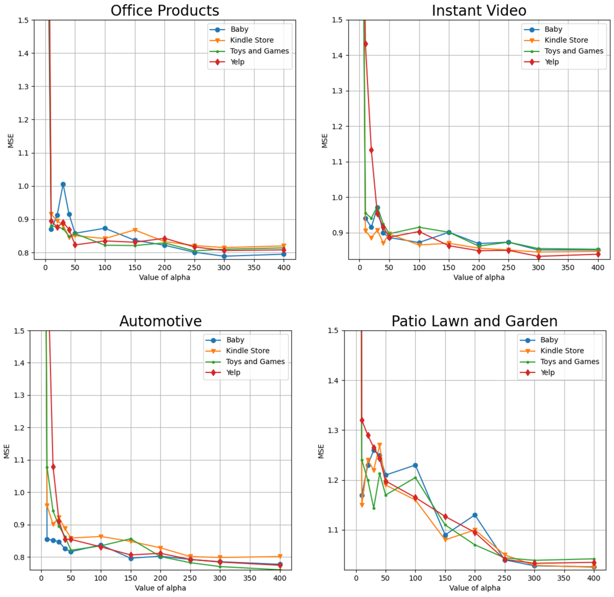


Figure 8: Convergence analysis w.r.t. MSE score for test data

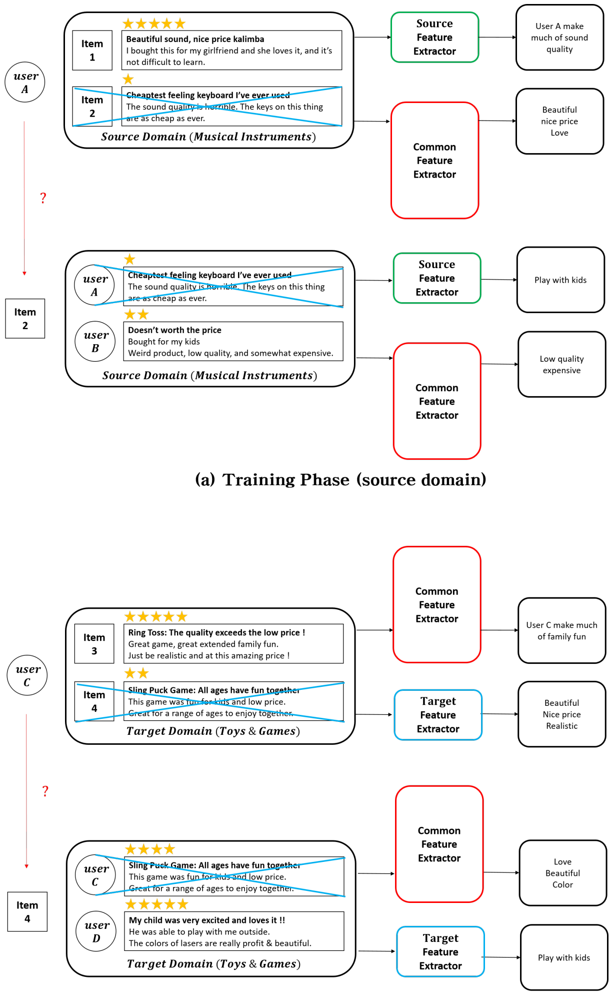
First, we start with α which denotes a weight hyper-parameter for a domain loss. Three feature extractors (FEs) are trained based on three losses. Generally, when α becomes large, e.g., 1.0, the FEs focus on the minimization of domain loss, while disregarding the reduction of classification loss (making an inaccurate prediction). On the contrary, when α gets lower, e.g., $\alpha = 0$, DaRE solely focuses on decreasing classification loss, which is independent of domain-aware feature extraction. Here, DaRE achieves the best performance if the value of α is 0.1.

Likewise, for β, the weight hyper-parameter affects three FEs and the encoder to generate similar embeddings between the individual review and aggregated reviews. In our experiments, β with 0.05 shows better performance compared to other values.

As can be seen, proportional to the value of α and β, the prediction error increases. Through a grid search, we systematically define parameters, $\alpha = 0.1$ with $\beta = 0.05$, which shows relatively higher performance compared to another value.

Convergence Analysis

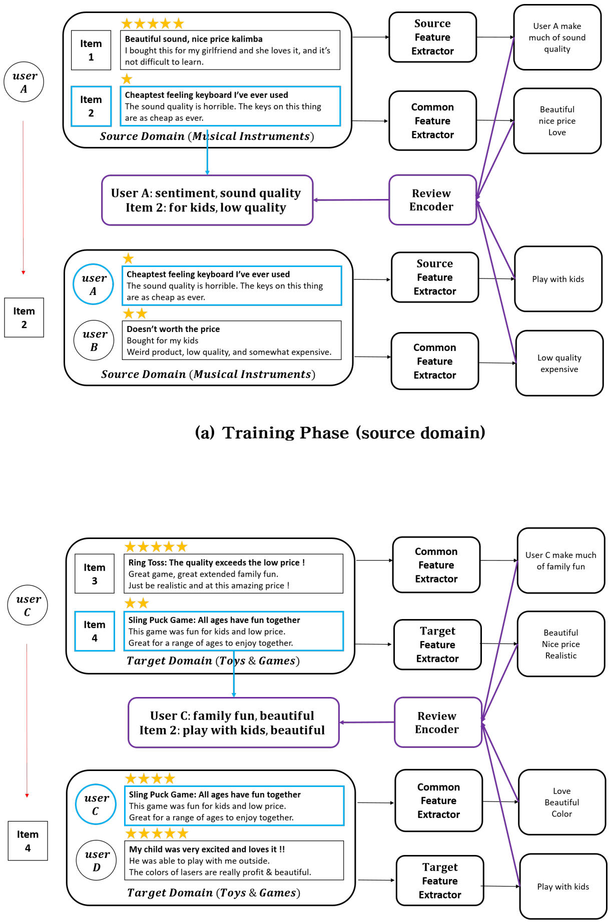
For a convergence analysis, we plot the MSE score based on four target domain datasets in Figure 8. Each figure demonstrates the MSE score of the target domain data based on four source domains, respectively. Here, we update the parameter of DaRE for 500 iterations. The x-axis and y-axis denote the number of iterations and MSE scores. At the beginning of training, DaRE shows relatively unstable performance due to the domain and encoding losses. As learning progresses, the domain and encoding losses decrease, leading to the convergence of parameters. Near 300 iterations, we can see that DaRE shows relatively stable results.


user A
Item 1
★★★★★
Beautiful sound, nice price kalimba
I bought this for my girlfriend and she loves it, and it's not difficult to learn.
Item 2
★
Cheaptest feeling keyboard I've ever used
The sound quality is horrible. The keys on this thing are as cheap as ever.
Source Domain (Musical Instruments)
Source Feature Extractor
Common Feature Extractor
User A make much of sound quality
Beautiful nice price Love
?
Item 2
user A
user B
★★
Doesn't worth the price
Bought for my kids
Weird product, low quality, and somewhat expensive.
Play with kids
Low quality expensive
Item 3
★★★★★
Ring Toss: The quality exceeds the low price !
Great game, great extended family fun.
Just be realistic and at this amazing price !
Item 4
★★
Sling Puck Game: All ages have fun together
This game was fun for kids and low price.
Great for a range of ages to enjoy together.
Target Domain (Toys & Games)
user C
Target Feature Extractor
User C make much of family fun
Beautiful Nice price Realistic
user D
★★★★
★★★★★
My child was very excited and loves it !!
He was able to play with me outside.
The colors of lasers are really profit & beautiful.
Love Beautiful Color
Play with kids

(a) Training Phase (source domain)

(b) Training Phase (target domain)

Figure 9: Example of domain-aware feature extraction. We assume two phases: training and inference, with two different domains: *Musical Instruments* and *Toys & Games* for cross-domain recommendation scenario


user A
Item 1
★★★★★
Beautiful sound, nice price kalimba
I bought this for my girlfriend and she loves it, and it's not difficult to learn.
Item 2
★
Cheaptest feeling keyboard I've ever used
The sound quality is horrible. The keys on this thing are as cheap as ever.
Source Domain (Musical Instruments)
Source Feature Extractor
Common Feature Extractor
User A make much of sound quality
Beautiful nice price Love
User A: sentiment, sound quality
Item 2: for kids, low quality
Review Encoder
Item 2
user A
user B
★★
Doesn't worth the price
Bought for my kids
Weird product, low quality, and somewhat expensive.
Play with kids
Low quality expensive
user C
Item 3
★★★★★
Ring Toss: The quality exceeds the low price !
Great game, great extended family fun.
Just be realistic and at this amazing price !
Item 4
★★
Sling Puck Game: All ages have fun together
This game was fun for kids and low price.
Great for a range of ages to enjoy together.
Target Domain (Toys & Games)
Target Feature Extractor
User C make much of family fun
Beautiful Nice price Realistic
User C: family fun, beautiful
Item 2: play with kids, beautiful
user D
★★★★
★★★★★
My child was very excited and loves it !!
He was able to play with me outside.
The colors of lasers are really profit & beautiful.
Love Beautiful Color
Play with kids

(a) Training Phase (source domain)

(b) Training Phase (target domain)

Figure 10: Example of review encoder. With user and item's previous reviews, the encoder assumes a real feedback that user will leave after purchasing an item

Proof of Equation 6, Domain Loss

The domain label d^s, d^t consists of binary values, $\{0, 1\}$ for source and target domain, respectively. Since a domain classification can be identified as a binary classification task, to retrieve a domain loss, we adopt binary cross-entropy loss for calculation.

First, we can induce a domain probability $[\widehat{d}^{c,s}_{u,i}, \widehat{d}^{c,t}_{u,i}]$ with interleaved common features $[O^{c,s}_{u,i}, O^{c,t}_{u,i}]$, respectively. The true label is $[0, 1]$, for source and target domain features. Here, we can define a binary cross entropy loss for common features as follows:

$$\mathcal{L}^{s,com}_{dom} = -\frac{1}{N_s} d^s \sum_{u,i\in D_s} log(\widehat{d}^{c,s}_{u,i}) - \frac{1}{N_s}(1-d^s)log(1-\widehat{d}^{c,s}_{u,i}) \ ,$$

$$\mathcal{L}^{t,com}_{dom} = -\frac{1}{N_t} d^t \sum_{u,i\in D_t} log(\widehat{d}^{c,t}_{u,i}) - \frac{1}{N_t}(1-d^t) \sum_{u,i\in D_t} log(1-\widehat{d}^{c,t}_{u,i}) \tag{18}$$

We can substitute domain labels d^s, d^t with $[0, 1]$, redefine Equation 18 as:

$$\mathcal{L}^{s,com}_{dom}, \mathcal{L}^{t,com}_{dom} = -\frac{1}{N_s} \sum_{u,i\in D^s} log(1-\widehat{d}^{s}_{u,i}), -\frac{1}{N_t} \sum_{u,i\in D^t} log(\widehat{d}^{t}_{u,i}) \tag{19}$$

Likewise, we can derive domain-specific losses with domain-specific features $\widehat{d}^{s}_{u,i}, \widehat{d}^{t}_{u,i}$ as follows:

$$\mathcal{L}^{s,spe}_{dom} = -\frac{1}{N_s} d^s \sum_{u,i \in D_s} log(\widehat{d}^{s}_{u,i}) - \frac{1}{N_s}(1-d^s) log(1-\widehat{d}^{c,s}_{u,i}) \ ,$$
$$\mathcal{L}^{t,spe}_{dom} = -\frac{1}{N_t} d^t \sum_{u,i \in D_t} log(\widehat{d}^{t}_{u,i}) - \frac{1}{N_t}(1-d^t) \sum_{u,i \in D_t} log(1-\widehat{d}^{c,t}_{u,i}) \tag{20}$$

Substitute domain labels d^s, d^t with $[0, 1]$, we can derive domain-specific losses $\mathcal{L}^{s,spe}_{dom}, \mathcal{L}^{t,spe}_{dom}$, which are defined in Equation 6.

Example of Domain-aware Feature Extraction

In Figure 9, we show an example of domain-aware feature extraction from a real-world benchmark dataset *Amazon*. The scenario assumes a training phase with source (Fig 9-(a), upper) and target (Fig 9-(b), lower) domain. The difference is that a common FE (red box) is shared across domains, while the source and target FEs (green and blue boxes) are domain-specific networks.

Taking Fig 9-(a) as an example, the objective is predicting a rating that a user A gives on item 2. Excluding individual review, $user\ A's\ review\ on\ item$ 2, the source and common extractors distillate latent of user and item respectively. Specifically, for user A in $MusicalInstruments$, a source FE captures domain-specific knowledge that she makes much of sound quality, while common FE extracts domain-common information like beautiful, and nice price. The analysis for item 2 follows the same mechanism.

To summarize, our model not only considers domain-shareable knowledge with common FE but also reflects domain-specific information through the source and target FE.

Example of Review Encoder

For the training of a review encoder, we utilize individual review that user A has written on item 2 (blue box) as another label. Taking Figure 10-(a) as an example, the review encoder (purple box) takes four types of inputs which are extracted from the source and common FEs. Then, the encoder generates a single output, which contains mixed information of user A and item 2. Here, the encoder is trained to infer an individual review, negative feedback of user A who takes sound quality into account. Likewise, another encoder in a target domain can be trained in a same manner.

REFERENCES

[1] Shankhadeep Banerjee, Samadrita Bhattacharyya, and Indranil Bose. 2017. Whose online reviews to trust? Understanding reviewer trustworthiness and its impact on business. *Decision Support Systems* 96 (2017), 17–26.

[2] Yang Bao, Hui Fang, and Jie Zhang. 2014. Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In *Twenty-Eighth AAAI conference on artificial intelligence.*

[3] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeshwar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and Devon Hjelm. 2018. Mutual information neural estimation. In *International Conference on Machine Learning.* PMLR, 531–540.

[4] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning* 79, 1 (2010), 151–175.

[5] Ruichu Cai, Zijian Li, Pengfei Wei, Jie Qiao, Kun Zhang, and Zhifeng Hao. 2019. Learning disentangled semantic representation for domain adaptation. In *IJCAI: proceedings of the conference*, Vol. 2019. NIH Public Access, 2060.

[6] Chong Chen, Min Zhang, Yiqun Liu, and Shaoping Ma. 2018. Neural attentional rating regression with review-level explanations. In *Proceedings of the 2018 World Wide Web Conference.* 1583–1592.

[7] Xinyang Chen, Sinan Wang, Bo Fu, Mingsheng Long, and Jianmin Wang. 2019. Catastrophic forgetting meets negative transfer: Batch spectral shrinkage for safe transfer learning. (2019).

[8] Xu Chen, Yongfeng Zhang, and Zheng Qin. 2019. Dynamic Explainable Recommendation based on Neural Attentive Models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 53–60.

[9] Zhongxia Chen, Xiting Wang, Xing Xie, Tong Wu, Guoqing Bu, Yining Wang, and Enhong Chen. 2019. Co-attentive multi-task learning for explainable recommendation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence.* AAAI Press, 2137–2143.

[10] Xin Dong, Jingchao Ni, Wei Cheng, Zhengzhang Chen, Bo Zong, Dongjin Song, Yanchi Liu, Haifeng Chen, and Gerard De Melo. 2020. Asymmetrical hierarchical networks with attentive interactions for interpretable review-based recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 7667–7674.

[11] Ali Mamdouh Elkahky, Yang Song, and Xiaodong He. 2015. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th International Conference on World Wide Web.* 278–288.

[12] Bairan Fu, Wenming Zhang, Guangneng Hu, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2021. Dual Side Deep Context-aware Modulation for Social Recommendation. In *Proceedings of the Web Conference 2021.* 2524–2534.

[13] Wenjing Fu, Zhaohui Peng, Senzhang Wang, Yang Xu, and Jin Li. 2019. Deeply fusing reviews and contents for cold start users in cross-domain recommendation systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 94–101.

[14] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research* 17, 1 (2016), 2096–2030.

[15] Lei Guo, Li Tang, Tong Chen, Lei Zhu, Quoc Viet Hung Nguyen, and Hongzhi Yin. 2021. DA-GCN: A Domain-aware Attentive Graph Convolution Network for Shared-account Cross-domain Sequential Recommendation. *arXiv preprint arXiv:2105.03300* (2021).

[16] Adeep Hande, Karthik Puranik, Ruba Priyadharshini, and Bharathi Raja Chakravarthi. 2021. Domain identification of scientific articles using transfer learning and ensembles. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining.* Springer, 88–97.

[17] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web.* 173–182.

[18] Guangneng Hu, Yu Zhang, and Qiang Yang. 2018. Conet: Collaborative cross networks for cross-domain recommendation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management.* 667–676.

[19] SeongKu Kang, Junyoung Hwang, Dongha Lee, and Hwanjo Yu. 2019. Semi-supervised learning for cross-domain recommendation to cold-start users. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management.* 1563–1572.

[20] Adit Krishnan, Mahashweta Das, Mangesh Bendre, Hao Yang, and Hari Sundaram. 2020. Transfer Learning via Contextual Invariants for One-to-Many Cross-Domain Recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.* 1081–1090.

[21] Bo Li, Yezhen Wang, Shanghang Zhang, Dongsheng Li, Kurt Keutzer, Trevor Darrell, and Han Zhao. 2021. Learning invariant representations and risks for semi-supervised domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.* 1104–1113.

[22] Jingjing Li, Erpeng Chen, Zhengming Ding, Lei Zhu, Ke Lu, and Heng Tao Shen. 2020. Maximum density divergence for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence* (2020).

[23] 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.* 331–339.

[24] Piji Li, Zihao Wang, Zhaochun Ren, Lidong Bing, and Wai Lam. 2017. Neural rating regression with abstractive tips generation for recommendation. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval.* 345–354.

[25] Huiting Liu, Lingling Guo, Peipei Li, Peng Zhao, and Xindong Wu. 2021. Collaborative filtering with a deep adversarial and attention network for cross-domain recommendation. *Information Sciences* 565 (2021), 370–389.

[26] Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. 2017. Cross-Domain Recommendation: An Embedding and Mapping Approach.. In *IJCAI*, Vol. 17. 2464–2470.

[27] David McAllester and Karl Stratos. 2020. Formal limitations on the measurement of mutual information. In *International Conference on Artificial Intelligence and Statistics.* PMLR, 875–884.

[28] Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems.* 165–172.

[29] Andriy Mnih and Russ R Salakhutdinov. 2008. Probabilistic matrix factorization. In *Advances in neural information processing systems*. 1257–1264.

[30] Xingchao Peng, Zijun Huang, Ximeng Sun, and Kate Saenko. 2019. Domain agnostic learning with disentangled representations. In *International Conference on Machine Learning*. PMLR, 5102–5112.

[31] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[32] Sainandan Ramakrishnan, Aishwarya Agrawal, and Stefan Lee. 2018. Overcoming language priors in visual question answering with adversarial regularization. *arXiv preprint arXiv:1810.03649* (2018).

[33] Noveen Sachdeva and Julian McAuley. 2020. How Useful are Reviews for Recommendation? A Critical Review and Potential Improvements. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1845–1848.

[34] Sungyong Seo, Jing Huang, Hao Yang, and Yan Liu. 2017. Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In *Proceedings of the eleventh ACM conference on recommender systems*. 297–305.

[35] Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2018. Multi-pointer co-attention networks for recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2309–2318.

[36] Xi Wang, Iadh Ounis, and Craig Macdonald. 2021. Leveraging Review Properties for Effective Recommendation. In *Proceedings of the Web Conference 2021*. 2209–2219.

[37] Xinghua Wang, Zhaohui Peng, Senzhang Wang, S Yu Philip, Wenjing Fu, and Xiaoguang Hong. 2018. Cross-domain recommendation for cold-start users via neighborhood based feature mapping. In *International conference on database systems for advanced applications*. Springer, 158–165.

[38] Xiangli Yang, Qing Liu, Rong Su, Ruiming Tang, Zhirong Liu, and Xiuqiang He. 2021. AutoFT: Automatic Fine-Tune for Parameters Transfer Learning in Click-Through Rate Prediction. *arXiv preprint arXiv:2106.04873* (2021).

[39] Wenhui Yu, Xiao Lin, Junfeng Ge, Wenwu Ou, and Zheng Qin. 2020. Semi-supervised collaborative filtering by text-enhanced domain adaptation. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2136–2144.

[40] Wenhui Yu, Huidi Zhang, Xiangnan He, Xu Chen, Li Xiong, and Zheng Qin. 2018. Aesthetic-based clothing recommendation. In *Proceedings of the 2018 world wide web conference*. 649–658.

[41] Fajie Yuan, Xiangnan He, Alexandros Karatzoglou, and Liguang Zhang. 2020. Parameter-efficient transfer from sequential behaviors for user modeling and recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1469–1478.

[42] Feng Yuan, Lina Yao, and Boualem Benatallah. 2019. DARec: deep domain adaptation for cross-domain recommendation via transferring rating patterns. *arXiv preprint arXiv:1905.10760* (2019).

[43] Hansi Zeng, Zhichao Xu, and Qingyao Ai. 2021. A Zero Attentive Relevance Matching Networkfor Review Modeling in Recommendation System. *arXiv preprint arXiv:2101.06387* (2021).

[44] Cheng Zhao, Chenliang Li, Rong Xiao, Hongbo Deng, and Aixin Sun. 2020. CATN: Cross-Domain Recommendation for Cold-Start Users via Aspect Transfer Network. *arXiv preprint arXiv:2005.10549* (2020).

[45] Lei Zheng, Vahid Noroozi, and Philip S Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. 425–434.

[46] Feng Zhu, Yan Wang, Chaochao Chen, Jun Zhou, Longfei Li, and Guanfeng Liu. 2021. Cross-domain recommendation: challenges, progress, and prospects. *arXiv preprint arXiv:2103.01696* (2021).