Single-Target and Dual-Target Cross-Domain Recommendation

by

Feng Zhu

A thesis submitted in fulfilment of

the requirements for the award of the degree

Doctor of Philosophy

from

Department of Computing

Faculty of Science and Engineering

MACQUARIE UNIVERSITY

Supervisor: Prof. Yan Wang

Associate Supervisor: Dr. Guanfeng Liu

Adjunct Supervisor: Dr. Chaochao Chen

September 2020

© Copyright by

Feng Zhu

September 2020

Statement of Candidate

I certify that the work in this thesis entitled **"Single-Target and Dual-Target Cross-Domain Recommendation"** has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree to any other university or institution other than Macquarie University.

I also certify that the thesis is an original piece of research and it has been written by me. Any help and assistance that I have received in my research work and the preparation of the thesis itself have been appropriately acknowledged.

In addition, I certify that all information sources and literature used are indicated in the thesis.

Feng Zhu

17 September 2020

To my parents and my girlfriend, who make me understand the true meaning of love.

Abstract

To address the data sparsity problem in recommender systems, cross-domain recommendation (CDR) has in recent years leveraged the relatively richer information from a richer (source) domain to only improve the recommendation performance in a sparser (target) domain with sparser information. Existing CDR approaches either directly replace a part of the latent representation of users/items in the sparser domain with the corresponding latent representation in the richer domain, or they map the latent representation of common users/items in the richer domain to fit those in the sparser domain.

First, finding an accurate mapping of the latent factors across domains is crucial for enhancing recommendation accuracy for CDR. However, this is a challenging task because of the complex relationships that exist between the latent factors of the source and the target domains or systems. To this end, this thesis proposes a deep framework for both cross-domain and cross-system recommendations (DCDCSR) based on matrix factorisation (MF) models and a fully connected deep neural network (DNN). Specifically, DCDCSR first employs the MF models to generate user and item latent factors and then employs the DNN to map the latent factors across domains or systems. More importantly, this approach considers the rating sparsity degrees of individual users and items in different domains or systems and uses them to guide the DNN training process for utilising the rating data more effectively.

Second, the existing CDR approaches are single-target approaches. However, each of the two domains may be relatively richer in certain types of information (e.g., ratings, reviews, user profiles, item details and tags). If such information can be leveraged well, it is thus possible to simultaneously improve the recommendation performance in both domains (i.e., dual-target CDR) rather than in a single-target domain

only. Thus, to achieve dual-target CDR, this thesis proposes a new framework for dual-target cross-domain recommendation (DTCDR). In the DTCDR framework, rating and multi-source content information are first extensively used to generate rating and document embeddings of users and items. Then, based on multi-task learning (MTL), an adaptable embedding-sharing strategy is designed to combine and share the embeddings of common users across domains, with which DTCDR can improve the recommendation performance on both richer and sparser (i.e., dual-target) domains simultaneously.

Third, inspired by DTCDR, this thesis attempts to further improve the recommendation performance in both domains. There are two new challenges: (1) how to generate more representative user and item embeddings, and (2) how to effectively optimise the user/item embeddings in each domain. To address these challenges, this thesis proposes a graphical and attentional framework, called GA-DTCDR. In the GA-DTCDR framework, two separate heterogeneous graphs are first constructed based on the rating and content information from the two domains to generate more representative user and item embeddings. Then, an element-wise attention mechanism is proposed for effectively combining the embeddings of common users learned from both domains. Both steps significantly enhance the quality of user and item embeddings and thus improve the recommendation accuracy in each domain.

All the above approaches proposed in this thesis have been validated and evaluated by theoretical analysis and extensive experiments conducted on real-world datasets. The experimental results demonstrate that the proposed methods significantly improve the recommendation accuracies in both the richer and sparser domains, and that these approaches outperform the state-of-the-art single-domain recommendation approaches, single-target CDR approaches, and dual-target CDR approaches in terms of recommendation accuracy.

Acknowledgments

First of all, I would like to express my sincere thanks to my supervisors, Prof. Yan Wang (principal supervisor), Dr. Guanfeng Liu (associate supervisor), and Dr. Chaochao Chen (adjunct supervisor), who helped me a lot during my doctoral study at the Macquarie University. Their professional suggestions and patient supervision let me find the correct way of my research. Over the past few years, I learned a lot when working with them, such as the rigorous attitude and dedicated spirit on scientific research. Especially to my principal supervisor Prof. Yan Wang, he selflessly spends so much time and energy to supervise my research. In a word, it is my great honour to have them as my supervisors at Macquarie University.

Second, I wish to express my thanks to my colleagues and the staff in the Department of Computing for their help. They brought me many valuable suggestions and a comfortable working environment during these years.

Most importantly, I would like to thank my parents, Zhaonian Zhu and Yongmei Liang. Their endless love, support, and encouragement make me have enough confidence and brave to accomplish this work.

Publications

This thesis is based on the research work I have completed with the help of my supervisors and other colleagues during my PhD program in the Department of Computing, Macquarie University between 2017 and 2020. Some parts of my research have been published/accepted by the following prestigious conferences:

[1] **Feng Zhu**, Yan Wang, Chaochao Chen, Guanfeng Liu, and Xiaolin Zheng: Graphical and Attentional Framework for Dual-Target Cross-Domain Recommendation, 29th International Joint Conference on Artificial Intelligence (IJCAI 2020), accepted in April 2020. (**research track, acceptance rate 12.6%, CORE2018**[1] **Rank A***)

[2] **Feng Zhu**, Chaochao Chen, Yan Wang, Guanfeng Liu, and Xiaolin Zheng: DTCDR: A Framework for Dual-Target Cross-Domain Recommendation, 28th ACM International Conference on Information and Knowledge Management (CIKM 2019), pages 1533-1542. (**research track, acceptance rate 19.4%, CORE2018 Rank A**)

[3] **Feng Zhu**, Yan Wang, Chaochao Chen, Guanfeng Liu, Mehmet A. Orgun, and Jia Wu: A Deep Framework for Cross-Domain and Cross-System Recommendations, 27th International Joint Conference on Artificial Intelligence (IJCAI 2018), pages 3711-3717. (**research track, acceptance rate 20.5%, CORE2018 Rank A***)

[1] CORE stands for Computing Research and Education Association of Australasia (`http://www.core.edu.au`).

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Significance

In the past couple of decades, recommender systems (RSs) have become a popular technique in many web applications — such as *MovieLens* (video sharing), *Amazon* (e-commerce) and *Facebook* (social networking) — and they provide suggestions of items to users so that users avoid facing the information overload problem [92]. Collaborative filtering (CF) has been proven to be the most promising technique in RSs [96]. The main goal of CF techniques is to recommend items to a user based on the observed preferences of other users whose historical preferences are similar to those of the target user [45]. However, in most real-world application scenarios, few users can provide ratings or reviews for many items [92] (i.e., data sparsity), which reduces the recommendation accuracy of matrix factorisation-based (MF-based) models [97, 4]. Almost all existing MF-based recommender systems suffer to some degree from this long-standing data sparsity problem, especially for new items or users (the cold-start problem). This problem may lead to over-fitting when training a CF-based model, which significantly reduces recommendation accuracy.

To address this data sparsity problem, a new trend has emerged in recent years that utilises the relatively richer information — such as observed ratings [58, 84, 85, 29, 42, 139, 71, 142], tags [106, 1, 47, 23], reviews [108], user/item information [19], semantic properties [24] and thumbs-up [99] — from the richer (source) domain to improve the recommendation accuracy in the sparser (target) domain [120, 25, 92].

Such approaches are termed *cross-domain recommendation* (CDR) [8]. Like CDR, cross-system recommendation (CSR) is also an effective solution for the data sparsity problem. CSR leverages the ratings or the knowledge derived from the source system to improve the recommendation accuracy in the target system, in which both systems are in the same domain [138]. For example, the Douban website's [1] RS can recommend books to users according to their movie reviews (i.e., CDR), since users in different domains are likely to have similar tastes. Additionally, the movie features derived from Netflix's system [2] can be transferred to Douban's system [138] (i.e., CSR) because both Netflix and Douban have the same domain of movie reviews, which is an example of CSR [138].

Existing CDR approaches can be classified into two groups: content-based transfer approaches and feature-based transfer approaches. *Content-based transfer* in CDR tends to link different domains by identifying similar content information, such as user profiles, item details [19, 8, 120], user-generated reviews [108] and social tags [1, 23]. In contrast, *feature-based transfer* [58, 84, 83, 85, 138, 137, 139, 132, 142, 131, 133] involves first training different CF-based models — such as singular value decomposition (SVD) [20], maximum-margin matrix factorisation (MMMF) [104], probabilistic matrix factorisation (PMF) [78], bayesian personalised ranking (BPR) [91], neural collaborative filtering (NCF) [38] and deep matrix factorisation (DMF) [123], to obtain user/item embeddings or patterns, and then transferring these embeddings through common or similar users/items across domains. In contrast to the content-based transfer approaches, feature-based transfer approaches typically employ machine learning techniques, such as transfer learning [137] and neural networks [71], to transfer knowledge across domains.

Additionally, all these existing CDR approaches only focus on how to leverage the source domain to help improve the recommendation accuracy in the target domain, not vice versa — namely, they are single-target CDR approaches. However, each of

[1] This webpage can be accessed from the following URL: https://www.douban.com
[2] Netflix can be accessed from the following URL: https://www.netflix.com

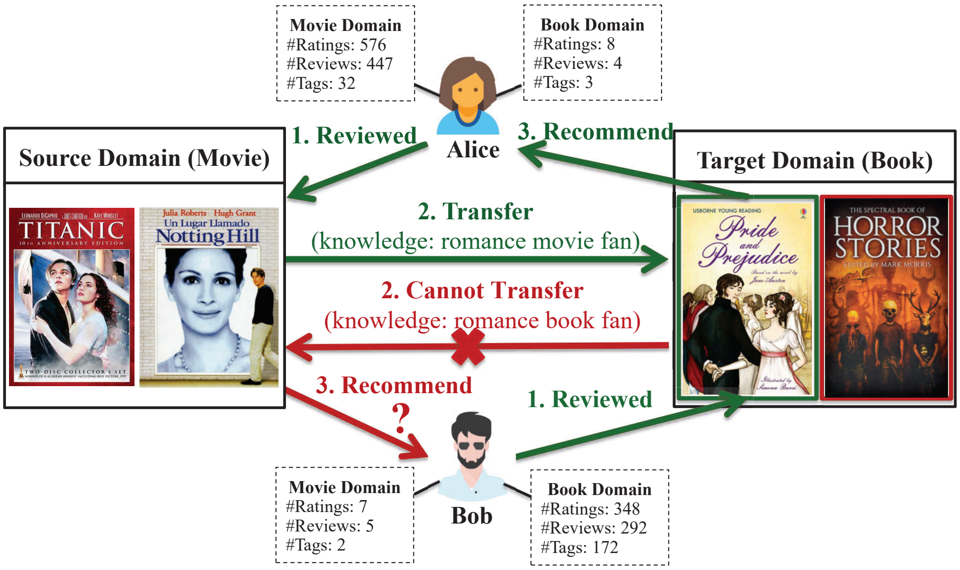


Figure 1.1: An example of a conventional single-target CDR (Movie → Book)

the two domains may be relatively richer in certain types of information (e.g., ratings, reviews, user profiles, item details and tags); if such information can be leveraged well, then it is thus possible to improve the recommendation performance in both domains simultaneously rather than only in a single target domain. This is also explained in the following example.

Motivating Example 1. Figure 1.1 depicts a conventional single-target CDR system. It contains two domains: a *movie domain* with relatively richer comments and a *book domain* with sparser comments. Essentially, the conventional CDR approaches transfer knowledge learned from the source *movie domain* to improve the recommendation accuracy in the target *book domain*, but not vice versa. To exemplify a typical case, suppose that Alice reviewed many movies (e.g., *Titanic* and *Notting Hill*) but that she reviewed only a few books. A conventional CDR system can recommend romance books (e.g., *Pride and Prejudice*) rather than *Horror Stories* to Alice since she is a fan of romance movies. To exemplify a special case, suppose that Bob reviewed many books and a few movies only. The conventional CDR system cannot make accurate movie recommendations for him because, in principle, the knowledge learned from the

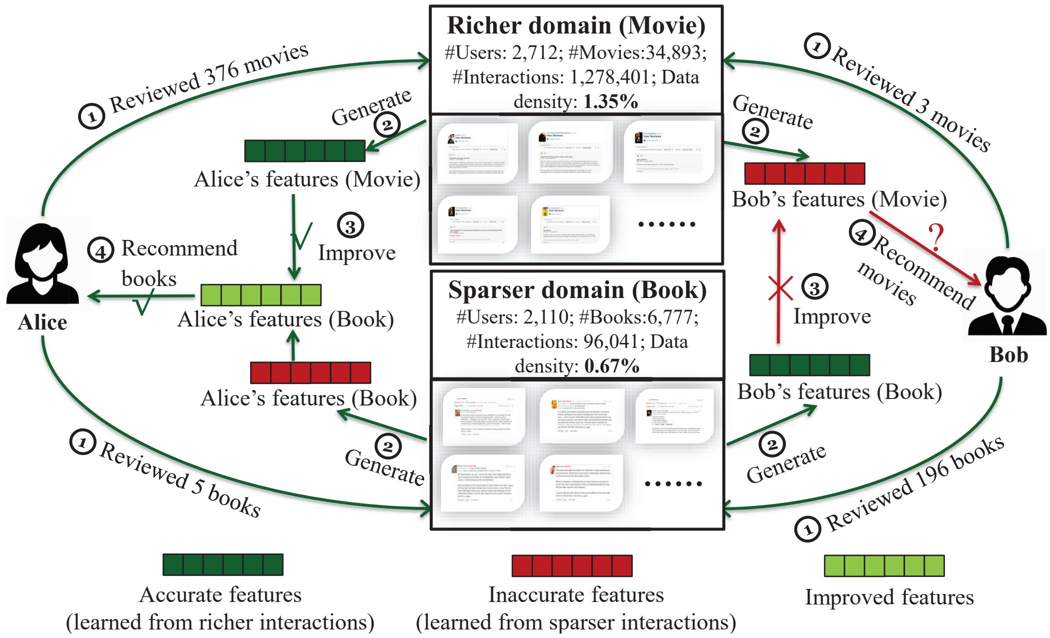


Figure 1.2: An example of a conventional single-target CDR: Movie (Richer) → Book (Sparser)

sparser domain is less accurate than that learned from the richer domain. This means that the conventional CDR system cannot work well by simply changing the transfer direction as from the sparser domain to the richer domain.

It can be intuited, based on the existing single-target CDR approaches, that this is a solution for dual-target CDR, by simply changing their transfer direction from "Richer→Sparser" to "Sparser→Richer". However, this idea — termed *Negative Transfer* [82] — does not work because, in principle, the knowledge learned from the sparser domain is less accurate than that learned from the richer domain; the recommendation accuracy in the richer domain is thus more likely to decline by simply and directly changing the transfer direction. Therefore, dual-target CDR demands novel and effective solutions. This motivation is further explained in the following example.

Motivating Example 2. Similar to Motivating Example 1, Figure 1.2 also depicts an example of a conventional single-target CDR system that contains two domains — movie (the richer domain) and book (the sparser domain) — including users, items (movies or books) and interactions (e.g., ratings, reviews, tags). The movie domain contains richer interactions than the book domain (1.35% v. 0.67% in data density). To exemplify a typical case (the majority of cases in the system), suppose that Alice reviewed 376 movies and five books. The two domains can generate her movie features and book features respectively by using feature representation models, such as CF models. In principle, Alice's movie features should be more accurate than her book features because richer interaction information can help train the feature representation models more effectively. Eventually, Alice's improved book features can be used to recommend matched books to her and thus improve the recommendation accuracy in the book domain. In contrast, to exemplify a different case (the minority of cases in the system), suppose that Bob reviewed only three movies and 196 books; Bob's book features would be more accurate than his movie features. However, the book features cannot be used to improve the movie features since the single-target CDR system can only leverage the information from the richer domain to improve the recommendation accuracy in the sparser domain. Therefore, motivated by the two types of cases described above, a novel dual-target CDR approach is critical and must be devised.

1.2 Challenges in Cross-Domain Recommendation

1.2.1 Accurate Mapping in Single-Target CDR

The first challenge of this thesis, **CH1**, is represented by the following question: *How can an accurate mapping of the latent factors across domains be found for enhancing recommendation accuracy?*

The common concept of the existing transfer-based transfer approaches in CDR is to map the latent factors obtained from a source domain (a relatively richer data source) to a target domain (a sparser data source) for improving recommendation accuracy. Therefore, accurately mapping the latent factors across domains is crucial for enhancing recommendation accuracy in CDR.

However, the existing transfer-based transfer approaches cannot effectively obtain an accurate mapping of the latent factors between the two domains. They either directly replace a part of the latent factors in the target domain with the corresponding latent factors in the source domain [139] (Category 1), or they map the latent factors of common users/items in the source domain to fit those factors in the target domain [71] (Category 2). The approaches in Category 1 ignore the complex relationship between the latent factors in the two domains, while the approaches in Category 2 only focus on the common users/items so that their relatively accurate latent factors in the source domain can be adjusted to fit the worse factors in the target domain, which is neither reasonable nor effective.

1.2.2 Feasible Dual-Target CDR Framework

The second challenge of this thesis, **CH2**, is represented by the following question: *How can a feasible framework for dual-target CDR be devised?*

The novel dual-target CDR problem faces a new challenge, one that has had no solutions reported in the literature — that is, how to devise a feasible framework for dual-target CDR. As an option, multi-task learning (MTL) has the potential for dual-target CDR because it aims to improve models' generalisation by leveraging the domain-specific information derived from the related recommendation tasks [94]. However, the existing MTL-based recommendation approaches [3, 69] cannot be efficiently applied to dual-target CDR because they heavily rely on the local feature representation and side information (additional information associated with users/items) from a single domain. Such features and information in the sparser domain may be too sparse to support dual-target CDR.

Additionally, multi-domain recommendation (MDR) is another potential option. However, the proposed MDR models in [135, 80, 85, 137] achieve different goals: they either focus on improving the recommendation accuracies of specific or common users selected from multiple domains, or they only improve the recommendation accuracy on a single target domain. None of these approaches can improve the recommendation accuracies of all users on multiple domains simultaneously. Therefore, the existing MDR models cannot serve for dual-target CDR directly.

1.2.3 More Representative Embedding

The third challenge of this thesis, **CH3**, is represented by the following question: *How can data richness and diversity be leveraged to generate more representative single-domain user/item embeddings for improving recommendation accuracy in both domains?*

Both traditional CF models (e.g., BPR [91]) and novel neural CF models (e.g., NeuMF [38] and DMF [123]) are based on the user-item relationship to learn user and item embeddings. However, most of these models ignore the user-user and item-item relationships, and can thus hardly enhance the quality of embeddings. To address the data sparsity problem, multi-source information (e.g., ratings, reviews, user profiles, item details and tags) derived from both domains should be leveraged to obtain more general user and item embeddings.

1.2.4 Embedding Optimisation

The fourth challenge of this thesis, **CH4**, is represented by the following question: *How can the user or item embeddings in each target domain be effectively optimised for improving recommendation accuracies in both domains?*

State-of-the-art dual-target CDR approaches either adopt fixed combination strategies (e.g., average-pooling, max-pooling and concatenation [141]) or they simply adapt the existing single-target transfer learning to dual-transfer learning [63]. However, none of these approaches can effectively combine the embeddings of common users/items, so it is difficult to achieve an effective embedding optimisation in each target domain.

1.3 Thesis Contributions

By targeting the significant challenges in CDR mentioned above, this thesis has three major contributions.

1. The first contribution is proposing a novel approach to generating benchmark factors that combine the features of the latent factors in both the source and target domains or systems. The latent factors in the target domain or system are then mapped to fit the benchmark factors. To the best of our knowledge, this process leads to a new category of transfer-based approaches for mapping latent factors across domains or systems — and this thesis's approach is the first in this novel category.

   The characteristics and contributions of this framework are summarised as follows:

   (a) Regarding **CH1**, this thesis proposes a **D**eep framework for both **C**ross-**D**omain and **C**ross-**S**ystem **R**ecommendations (DCDCSR), which employs MF models and a fully connected deep neural network (DNN).

   (b) This thesis employs MF models to generate user and item latent factors. When generating benchmark factors, fine-grained sparsity degrees of individual users and items are considered to combine the latent factors learned from both the source and target domains or systems, which can effectively utilise more rating data in the two domains or systems.

   (c) The DNN is employed to accurately map the latent factors in the target domain or system to fit the benchmark factors, which can improve recommendation accuracy.

   (d) The extensive experiments conducted on three real-world datasets demonstrate that the DCDCSR framework outperforms the state-of-the-art approaches and that it clearly improves the recommendation accuracy for both CDR and CSR.

2. The second contribution is proposing a new framework for dual-target CDR. To the best of our knowledge, this research is the first work in the literature to propose the novel problem of dual-target CDR and provide a solution for it. The relevant characteristics and contributions of this framework are summarised as follows:

   (a) Regarding **CH2**, a novel framework is proposed for **D**ual-**T**arget **CDR** (DTCDR), which can leverage the data richness and diversity of dual domains, share the knowledge of common users across domains, and improve the recommendation accuracies for all users in both domains simultaneously.

   (b) Regarding **CH3**, multi-source text information (reviews, tags, user profiles and item details) is considered to generate the document embeddings of users and items by using Doc2Vec, as well as to optimise two rating embedding models (i.e., NeuMF and DMF) to generate the rating embeddings of users and items. Then, based on MTL, an effective embedding-sharing strategy is designed and three representative combination operators are chosen (i.e., Concatenation, Max-Pooling and Average-Pooling) to respectively combine the text and rating embeddings of common users. They can synthesise these embeddings in diverse ways and make the DTCDR framework adaptable to different scenarios.

   (c) The extensive experiments conducted on real-world Douban and MovieLens datasets demonstrate that the DTCDR approach significantly outperforms the state-of-the-art single-domain (SDR) and CDR approaches in terms of recommendation accuracy.

3. The third contribution is proposing a novel graphical and attentional approach for dual-target CDR. The characteristics and contributions of this work are summarised as follows:

(a) Regrading **CH2**, this thesis proposes a **G**raphical and **A**ttentional framework for **D**ual-**T**arget **C**ross-**D**omain **R**ecommendation (GA-DTCDR), which can leverage the data richness and diversity of dual domains (e.g., ratings, reviews and tags), share the knowledge of common users across domains and make recommendations in both domains.

(b) Regrading **CH3**, a heterogeneous graph is constructed that considers not only user-item relationships (based on ratings) but also user-user and item-item relationships (based on content similarities). With this heterogeneous graph, a graph embedding technique (i.e., Node2vec) is then applied to generate more representative single-domain user and item embeddings for accurately capturing user and item features.

(c) Regarding **CH4**, an element-wise attention mechanism is proposed to intelligently and effectively combine the embeddings of common users learned from both domains. This mechanism trains two separate element-wise attention networks for the two target domains respectively, which can significantly enhance the quality of user embeddings and thus improve the recommendation accuracy in both domains simultaneously.

(d) Extensive experiments are conducted on four real-world datasets and demonstrate that the GA-DTCDR approach significantly outperforms the best-performing baselines in all cases by an average of 8.46% in terms of recommendation accuracy.

1.4 Thesis Roadmap

The remaining thesis chapters are structured as follows.

Chapter 2 begins with a comprehensive literature review on SDR, single-target CDR and dual-target CDR. It also presents a brief literature review on other highly relevant fields such as MTL, graph embedding and attention mechanism.

Chapter 3 presents the DCDCSR framework, based on MF models and a fully connected DNN. This chapter includes our paper that was published at IJCAI 2018 [142].

Chapter 4 presents the DTCDR framework that can improve the recommendation performance in both source and target domains simultaneously. It includes our paper that was published at CIKM 2019 [141].

Chapter 5 presents the GA-DTCDR framework that can leverage the data richness and diversity of dual domains, share the knowledge of common users across domains and make recommendations in both domains. This chapter includes our paper that was accepted by IJCAI 2020 in April 2020.

Finally, Chapter 6 concludes the research in this thesis and offers directions for future research opportunities.

Chapter 2

Literature Review

In this chapter, according to recommendation problems, the related literature in three main categories mentioned in Sections 2.1, 2.2 and 2.3 are reviewed respectively: (1) single-domain recommendation (SDR), (2) single-target cross-domain recommendation (STCDR) and (3) dual-target cross-domain recommendation (DTCDR). Additionally, according to the techniques we employ in this thesis, we also briefly review the related literature about (4) multi-task learning (MTL), (5) graph embedding and (6) attention mechanism. In particular, because multi-task learning is employed for the first dual-target CDR framework (i.e., DTCDR), this chapter reviews the related literature about MTL in Section 2.4. To further improve the recommendation accuracies in both domains simultaneously, a new dual-target CDR solution is proposed by using a graph embedding technique to learn the content embedding of users and items, and an attention mechanism to combine common users' or items' embedding. Therefore, this chapter also reviews the related literature about graph embedding in Section 2.5 and attention mechanism in Section 2.6.

This chapter is organised as follows:

- Section 2.1 introduces the existing SDR approaches, including rating-based approaches and content-based approaches.

- Section 2.2 introduces the existing single-target CDR approaches, including content-based transfer approaches and feature-based transfer approaches.

- Section 2.3 introduces the existing dual-target CDR approaches.

- Section 2.4 introduces related MTL approaches.
- Section 2.5 introduces the related approaches in graph embedding.
- Section 2.6 introduces the related attention mechanism.
- Section 2.7 presents a summary of the existing studies.

2.1 Single-Domain Recommendation

According to the focus of this thesis, the existing SDR approaches are summarised in two groups: rating-based approaches and content-based approaches. There are many SDR approaches in the literature; but this thesis only reviews those that are highly relevant to its focus. For a clear comparison, these approaches are listed in detail in Tables 2.1 and 2.2.

2.1.1 Rating-Based SDR

Conventional single-target recommender approaches have largely focused on making item recommendations that are based on observed ratings in a single domain. These approaches are thus classified as rating-based single-target recommendation (rating-based SDR). These rating-based approaches can be generally classified into two broad categories according to different techniques, i.e., MF-based approaches [104, 7, 78, 91] and neural network-based (NN-based) approaches [18, 38, 123]. The MF-based approaches opt for learning a linear relationship between users and items, and their goals are to minimise the square or ranking loss between the observed and predicted ratings. In contrast, the NN-based approaches apply a DNN such as multi-layer perceptron (MLP) to learning a non-linear user-item interaction function and their goals are to minimise the loss between the observed and predicted interactions derived from ratings. These MF and NN-based approaches are introduced in the following sections.

Table 2.1: The comparison of existing SDR approaches (Part 1)

Category		Representative approaches	Technology adoption or basic idea
Rating-Based	MF-Based	MMMF – Srebro et al. [104]	Low-norm factorisation
		Bell et al. [7]	Singular value decomposition (SVD) [31]
		Koren et al. [52]	SVD
		Takacs et al. [107]	SVD
		PMF – Mnih et al. [78]	Gaussian distribution
		MF – Koren et al. [53]	Matrix factorisation
		RankBoost – Freund et al. [26]	Rating ranking minimisation
		RankNet – Burges et al. [10]	Rating ranking minimisation
		BPR – Rendle et al. [91]	Rating ranking minimisation
		RankSVM – Chapelle et al. [14]	Rating ranking minimisation
	NN-Based	Cheng et al. [18]	Wide & deep learning
		NeuMF – He et al. [38]	Wide & deep learning
		DMF – Xue et al. [123]	Deep learning

2.1.1.1 Matrix Facotrisation-Based Approaches

First, inspired by large-margin linear discrimination, Srebro et al. in [104] proposed a maximum-margin matrix factorisation (MMMF) approach to learning a fully predicted rating matrix to fit the observed rating matrix by minimising a trace norm and maximising the corresponding predictive margin. MMMF tends to apply low-norm factorisations rather than low-rank factorisations, which can avoid convex optimisation problems.

Then, based on the common idea of singular value decomposition (SVD) [31], the

Table 2.2: The comparison of existing SDR approaches (Part 2)

Category	Representative approaches	Technology adoption or basic idea
Content-based	Mooney et al. [79]	Information extraction & text categorisation
	LDA – Blei et al. [9]	Topic modelling
	CTR – Wang et al. [116]	factorisation &topic modelling
	HFT – Mcauley et al. [74]	factorisation &topic modelling
	TopicMF – Bao et al. [6]	factorisation &topic modelling

approaches proposed in [7, 52, 107] map both items and users into the same latent factor space. These latent factors can represent implicit user preferences and item features respectively, and the inner product of user latent factors and item latent factors can be treated as the predicted ratings from the users to the items.

Next, in [78], Mnih et al. proposed a probabilistic matrix factorisation (PMF) model which scales with the number of observed ratings. This model performs well on the large Netflix dataset. PMF is a probabilistic model with Gaussian observation noise, and its core idea is to maximise the conditional distribution over the observed ratings.

In [53], Koren et al. proposed an MF approach for RSs that won the Netflix Prize competition in 2009. Unlike the classic nearest-neighbour approaches for item recommendations, the proposed MF can leverage additional information such as implicit feedback, temporal effects and confidence levels, which can significantly improve the recommendation performance.

Unlike the above conventional rating-oriented approaches, ranking-oriented approaches, such as RankBoost [26], RankNet [10], BPR [91] and RankSVM [14], tend to optimise a loss function defined on users' pairwise preferences, as well as minimise the loss between the observed rating rankings and the predicted rating rankings.

2.1.1.2 Neural Network-Based Approaches

In [18], Cheng et al. proposed a joint framework to train both linear models and deep neural networks for single-target recommendation. This Wide & Deep learning can combine the advantages of memorisation and generalisation, which significantly improves the conventional wide-only and deep-only models.

Recently, neural matrix factorisation (NeuMF) [38] is a novel framework that replaces the dot product of the traditional MF with a non-linear relation (a neural network). Since the traditional neural network (i.e., MLP) [105], has difficulty in capturing low-rank relations, NeuMF combines MF and MLP into one model that can represent both wide and deep embeddings of users and items by learning implicit feedback.

Unlike NeuMF, deep matrix factorisation (DMF) [123] considers both implicit and explicit feedback. The core idea of DMF is evaluating the cosine similarities between user and item latent factors learned by their corresponding ratings. The cosine similarity between a user and an item can be treated as the predicted rating from the user to the item. Additionally, to consider both implicit and explicit feedback in the loss function, DMF improves the loss function of NeuMF by a normalised cross-entropy loss.

2.1.2 Content-Based Single-Domain Recommendation

Apart from rating-based SDR approaches, there are also content-based approaches [79, 9, 74, 65, 6] that focus on modelling both observed ratings and content information. The collaborative topic regression (CTR) model proposed in [116] is a major breakthrough for article (or citation) recommendation, which tends to combine the advantages of traditional CF and topic modelling. The models in [74, 6] attempt to combine the latent factors learned from ratings with the latent review topics learned from contents by the topic models, which can explore more prior knowledge on users and items. These content-based SDR approaches are introduced below.

To predict the unknown rating from a target user to an item, conventional CF-based approaches mainly focus on the preferences of the user's neighbours. In contrast to these conventional CF-based approaches, Raymond et al. in [79] proposed a content-based SDR approach that uses content information about an item itself to make book recommendations. This approach is based on information extraction and text categorisation.

In [9], Blei et al. proposed the latent Dirichlet allocation (LDA), a generative probabilistic model for text information that can generate the latent topics of users and items. LDA has become a basic technique for modelling latent topics in many content-based RSs. The basic concept of LDA is that the text information of users and items is represented as random mixtures over latent topics.

In [116], Wang et al. proposed the collaborative topic regression (CTR) approach for recommending scientific articles to users. This approach combines the benefits of traditional latent factor models and probabilistic topic models. Based on latent factor models, CTR can use the information from other users' libraries and recommend matched known articles to a target user. Additionally, based on topic modelling, CTR can generate the latent representation of the article and recommend matched unknown articles that have similar content to others that a target user likes.

In [74], McAuley et al. proposed the hidden factors as topics (HFT) approach, a content-based approach that combines latent factors with latent review topics. Specifically, HFT extracts highly interpretable labels for latent factors learned by latent factor models. These labels can help judge whether the latent factors are accurate or not.

To address the data sparsity problem in a single domain, Bao et al. in [6] proposed TopicMF, an MF model that considers both the ratings and the review texts. They applied a biased MF for rating prediction and a topic modelling technique for latent topic modelling. They then combined the latent factors learned by the biased MF and the latent topics learned by the topic modelling. Compared with conventional latent factor models, TopicMF can leverage rich review information and thus improve recommendation accuracy.

2.1.3 Single Domain Recommendation: A Summary

Existing rating-based SDR approaches are mainly based on the notion of CF and leverage the known rating information to learning a linear (e.g., MF) or non-linear (e.g., neural networks) relation between users and items. These approaches can work well for recommending known items. Conventional rating-based SDR approaches tend to employ factorisation methods (e.g., low-norm factorisation, SVD and MF) to generate the latent factors of users and items. In addition to these factorisation-based approaches are ranking-oriented approaches, which focus on minimising the rating ranking loss. Recently, based on Wide & Deep learning, some novel rating-based SDR approaches focus on learning a non-linear relationship between users and items.

Additionally, to improve the recommendation accuracy for unknown items, content-based SDR approaches are proposed to leverage the content information of users and items so that the content similarities between two items or two users can be measured. The content similarities can help recommend matched items to a target user. These approaches tend to employ information extraction techniques and topic modelling techniques to leverage useful content information.

However, all these SDR approaches are constrained by limited data from a single domain, regardless of whether the data is derived from rating information or content information. This means that there are some difficulties in using these approaches to further solve the data sparsity problem — which is the fundamental motivation of proposing CDR.

2.2 Single-Target Cross-Domain Recommendation

Most of existing single-target CDR approaches tend to leverage useful information from the source domain to the target domain. According to transfer strategies, these single-target CDR approaches are reviewed in two categories: content-based transfer and feature-based transfer.

- **Content-based transfer.** These approaches first create links based on the common contents, e.g., user/item attributes [8, 57], social tags [106, 1, 47, 101, 23], semantic properties [24, 55, 130], thumbs-up [99], text information [110, 108], metadata [95], browsing or watching history [21, 48]. Then they transfer user preferences or item details across domains.

- **Feature-based transfer.** These approaches employ some classical machine learning models — such as MTL [102, 3, 69], transfer learning [59, 84, 138, 61, 115, 139, 89, 132, 40, 98, 37, 131, 133, 41, 73, 43, 62], clustering [90, 88, 22, 119], reinforcement learning [66], deep neural networks [46, 71, 142, 36, 27, 67], relational learning [103] and semi-supervised learning [49], to map or share features, e.g., user/item latent factors and rating patterns [58, 29, 37, 128], learned by MF models across domains.

Additionally, [135, 80, 85, 137] proposed multi-domain models, but they either tend to make recommendations for specific or common users that are selected from domains, or only for the users in the target domain. In contrast, this thesis's proposed DTCDR and GA-DTCDR frameworks aim to achieve a different goal — making recommendations for all users in both the source and target domains.

For a clear comparison, these single-target CDR approaches are listed in detail in Tables 2.3 - 2.7.

2.2.1 Content-Based Transfer

Existing content-based transfer approaches tend to leverage different types of content information from the source domain to address the data sparsity problem in the target domain. The content information is treated as a bridge that links the two domains. According to the types of content information, we classify these content-based transfer approaches into the following seven sub-categories, i.e., user/item attributes, social tags, semantic properties, thumbs-up, text information, metadata, browsing or watching history.

Table 2.3: The comparison of existing single-target CDR approaches (Part 1)

Category		Representative approaches	Technology adoption or basic idea
Content -based transfer	User/item attributes	Berkovsky et al. [8]	Multi-source information
		CLARE – Leung et al. [57]	User-item & item-item relationships
	Social tags	Szomszor et al. [106]	Co-occurrence sub-graph
		Abel et al. [1]	Profile semantic enhancement
		Kaminskas et al. [47]	Tag similarity
		TagCDCF – Shi et al. [101]	Tag similarity
		TagGSVD++ – Fernandez et al. [23]	Rating-tag similarity
	Semantic properties	Fernandez et al. [24]	Weighted directed acyclic graph
		Kumar et al. [55]	Semantic similarity
		Zhang et al. [130]	Semantic correlation
	Thumbs-up	Shapira et al. [99]	Preference similarity

2.2.1.1 User/Item Attributes

In [8], Berkovsky et al. first proposed CDR to address the data sparsity problem for CF recommenders. In this study, the authors leveraged four types of user modelling data: user models that were learned from the source domain, lists of the neighbours, similarity degrees between the active user and the other users, and predicted ratings that were generated from the source domain. The authors then imported and aggregated the vectors of users' ratings that were learned from different application domains to improve the prediction accuracy.

Further, in [57], Leung et al. proposed Cross-Level Association RulEs, a hybrid CDR approach that addresses the cold-start problem. They integrated the content information from the items of two different domains into the conventional CF models. Specifically, they first applied a preference model to represent both user-item and

Table 2.4: The comparison of existing single-target CDR approaches (Part 2)

Category	Representative approaches	Training data	Technology adoption or basic idea
Content -based transfer	Text information	CTL – Tang et al. [110]	Topic modelling
		Tan et al. [108]	Topic modelling & transfer learning
		Sahebi et al. [95]	User similarity
	Browsing or watching history	Elkahky et al. [21]	Multi-view learning
		Kanagawa et al. [48]	Unsupervised domain adaptation

item-item relationships, and then, based on this preference model, they made recommendations for cold-start items.

2.2.1.2 Social Tags

The two approaches described above tend to apply the common user/item attributes from the two domains to improve recommendation accuracy. Apart from user/item attributes, social tags also helpfully address the data sparsity problem in CF because social tags are an important source of cross-domain user preferences. In [106], Szomszor et al. proposed an approach for recommending a set of social tags to annotate a bookmarked document. In this study, the authors processed the bookmark textual contents and then abstracted a set of keywords from these contents. Finally, based on the keywords (tags), they built a co-occurrence sub-graph to make recommendations. In [1], Abel et al. performed a profile semantic enhancement process by grouping tags into WordNet categories. The experimental results revealed that the integrated profiles from two different domains can improve recommendation quality. In [47], Kaminskas et al. proposed a location-adapted music recommendation approach that could apply the emotional tags of users to two different domains (i.e., music and places of interest [POIs]). This approach matches music tracks and POIs by considering their tag similarities and then making music recommendations according to users' POIs. In [101], Shi et al. proposed the tag-induced cross-domain collaborative filtering (TagCDCF) approach, which leverages social tags to improve recommendation accuracy. TagCD-CF uses tag-based similarities to link two domains, and thus does not require common users or items. Additionally, in [23], Fernandez et al. proposed TagGSVD++, a content-based CDR approach that shares the social tags across domains. Based on SVD++ (a variety of SVD) [52], TagGSVD++ can generate more representative latent factors of users and items that can better capture the effects of social tags on ratings.

2.2.1.3 Semantic Properties

In addition to user/item attributes and social tags, semantic properties are also important for improving the recommendation accuracy in CDR scenarios. Fernandez et al. proposed a semantic-based framework for CDR [24] that can automatically extract useful information from two different domains and then link the concepts of the two domains by using a weighted directed acyclic graph. In the study's experiments, this approach could effectively recommend music artists to users by identifying the related POIs in the source domain. In [55], Kumar et al. proposed a semantic clustering-based approach (i.e., SCD) for CDR, in which a common semantic space is shared across domains. SCD applies a topic model to generate user/item latent topics and applies an ontology model to measure semantic similarities between textual words from different domains.

Recently, Zhang et al. [130] leveraged tag information as a bridge to link two domains. They considered the semantic correlations from the overlapping tags and applied these correlations to make recommendations. Specifically, they applied the word2vec model to learning the latent representations of tags — and these representations can be treated as a bridge for sharing across domains.

2.2.1.4 Thumbs-up

In [99], the thumbs-up function is important auxiliary information for improving the recommendation performance in CDR scenarios. In their study, Shapira et al. leveraged auxiliary social data (Facebook preference data; users' favourite items) to improve the recommend accuracy on other domains. For the cold-start users in these experiments, which were only based on Facebook social data, their approach also achieved a pretty good performance (i.e., it was no less accurate than what was obtained from user ratings).

2.2.1.5 Text Information

In [110], Tang et al. proposed a cross-domain topic learning (CTL) model for recommending research articles to users. CTL consolidates the collaborations by topic layers rather than by author layers because author connections between two domains are rare. Additionally, CTL generates the latent topics in the source and target domains separately, and it only models relevant topics between the two domains.

In [108], Tan et al. utilised text information from the source domain to improve the recommendation accuracy in the target domain. This approach is based on LDA [9], with its core concept being the transfer of user interests across domains. Tan et al. extracted documents (text information) from different domains and then combined the text information with ratings. They modelled user interests in the common topic space and, based on this space, could recommend matched items to users in the target domain.

2.2.1.6 Metadata

In [95], Sahebi et al. leveraged user metadata to improve the recommendation accuracy in CDR scenarios by proposing a generic framework for content-based CDR. Compared with traditional CDR approaches, these authors introduced user-based domains rather than item-based domains, which signifies that they distinguished the different domains according to the types of users (e.g., young and old) rather than the types of items (e.g., movie and book).

2.2.1.7 Browsing or Watching History

Based on multi-view learning, Elkahky et al. [21] leveraged the auxiliary web-browsing history and search queries from a source domain to improve the recommendation accuracy in a target domain. This approach involves jointly learning item features from different domains and user features by using a multi-view deep-learning model.

Additionally, Kanagawa et al. posited a content-based CDR framework for cold-start users [48]. This framework does not require common users or items to be a bridge linking the two domains. The general concept of this framework is treating the recommendation task as a classification task, with the classification task being an instance of unsupervised domain adaptation.

2.2.2 Feature-Based Transfer

Feature-based transfer is the most popular category of the existing CDR approaches, and its general description involves transferring knowledge (e.g., user/item latent factors and rating patterns) from the source domain to help the target domain. According to the techniques, the features-based transfer approaches are classified into the following five subcategories: MTL, transfer learning, clustering, deep neural networks and others.

2.2.2.1 Multi-Task Learning-Based Approaches

MTL is a classical technique used for single-target CDR. Based on MTL, RSs can leverage the data from multiple views or sources and the knowledge from multiple learning tasks to improve the recommendation accuracy in the target domain. This section briefly introduces the representative works in this field below.

To address the longstanding data sparsity problem in CF, Singh et al. suggested a

Table 2.5: The comparison of existing single-target CDR approaches (Part 3)

Category		Representative approaches	Training data	Technology adoption or basic idea
Feature -Based Transfer	Multi-task learning	CMF – Singh et al. [102]	Ratings & item details	Multiple relations
		LMF – Agarwal et al. [3]	Multiple contexts	multi-task learning
		Lu et al. [69]	Ratings	Multi-task learning
	Transfer learning	RMGM – Li et al. [59]	Ratings	Transfer learning
		CST – Pan et al. [84]	Heterogeneous feedback	Principle coordinates
		Zhao et al. [138]	Ratings	Active transfer learning
		Li et al. [61]	Ratings	Transfer learning
		Wang et al. [115]	Mails	Mailing list similarity
		Zhao et al. [139]	Ratings	Active transfer learning
		Rafailidis et al. [89]	Ratings	Transfer learning
		CIT – Zhang et al. [132]	Ratings	Consistent information
		KerKT – Zhang et al. [131]	Ratings	Domain adaptation & diffusion kernel completion
		ProbKT – Zhang et al. [133]	Ratings	Feature combination
		CoNet – Hu et al. [40]	Ratings	Neural networks
		Shang et al. [98]	Ratings	Transfer learning

collective matrix factorisation (CMF) framework that was based on relational learning in their study [102]. They considered encoding users' ratings, movies' genres and actors' roles in different relations; they could thus improve the predictive accuracy in relation by leveraging the information from another relation.

Further, Agarwal et al. proposed a localised matrix factorisation (LMF) approach in [3]. LMF combines multiple contexts to improve the recommendation accuracy in the target domain. In LMF, the local factors learned from different contexts are linked to a shared global model, rather than the different local factors of an entity (a

Table 2.6: The comparison of existing single-target CDR approaches (Part 4)

Category		Representative approaches	Training data	Technology adoption or basic idea
Feature -Based Transfer	Transfer learning	TMH – Hu et al. [41]	Ratings & text	Transfer learning & memory networks
		Manotumruksa et al. [73]	Ratings	Transfer learning
		LSCD – Huang et al. [43]	Ratings	Transfer learning
	Clustering	PCLF – Ren et al. [90]	Ratings	Clustering
		JCSL – Rafailidis et al. [88]	Ratings	User clustering
		C^3R – Farseev et al. [22]	Ratings	Clustering
		CDIE-C – Wang et al. [119]	Ratings	Clustering
	Deep neural networks	Jaradat et al. [46]	Ratings	Textual input relations
		EMCDR – Man et al. [71]	Ratings	Linear matrix translation & MLP
		He et al. [36]	Ratings	Bayesian neural networks
		RC-DFM – Fu et al. [27]	Ratings & content	Stacked Denoising Autoencoders
		ACDN – Liu et al. [67]	Ratings	Aesthetic preferences

user or an item) being enforced to be the same. This approach is an EM algorithm that contains two steps — the E-step, which involves two main operations (bottom-up filtering and top-down smoothing), and the M-step, in which the algorithm mainly focuses on parameter estimating.

Additionally, Lu et al. proposed a CDR approach in [69] that can predict unknown ratings, as well as explain the recommendation results. With the help of combined MF, this approach can predict unknown ratings more accurately, and with the help of an adversarial sequence, it can apply sequence learning to explain the recommendation

Table 2.7: The comparison of existing single-target CDR approaches (Part 5)

Category		Representative approaches	Training data	Technology adoption or basic idea
Feature -Based Transfer	Others	TCB – Liu et al. [66]	Ratings	Reinforcement learning
		Sopchoke et al. [103]	Ratings	Relational learning
		SSCDR – Kang et al. [49]	Ratings	Semi-supervised learning
		CBT – Li et al. [58]	Ratings	Transfer learning for rating patterns
		Li et al. [60]	Ratings	Bayesian latent factor models & interest Drift
		Gao et al. [29]	Ratings	Transfer learning for rating patterns
		CDTF – Hu et al. [42]	Explicit and implicit feedback	Triadic relation (user-item-domain)
		Loni et al. [68]	Ratings	Interaction patterns
		He et al. [37]	Ratings	Transfer learning
		Ma et al. [70]	Item sequences	Sequential recommendations
		NATR – Gao et al. [28]	Items' information	Privacy security
		DARec – Yuan et al. [128]	Ratings	Transfer learning for rating patterns

generation.

2.2.2.2 Transfer Learning-Based Approaches

Transfer learning is the most popular technique in single-target CDR. Based on transfer learning, there are many approaches that are proposed to achieve single-target CDR. Next, this thesis will briefly introduce representative and state-of-the-art works.

In [59], and based on transfer learning, Li et al. suggested a rating-matrix generative model (RMGM) for CDR. RMGM shares a common implicit cluster-level rating matrix that can pool the rating data from different domains. Based on this shared rating matrix, RMGM can predict unknown ratings and thus recommend matched items to users. The experimental results indicated that this approach can gain auxiliary useful knowledge from other domains to improve recommendation performance.

Pan et al. proposed the coordinate system transfer (CST) in [84], a new transfer learning-based framework for CDR. In this study, Pan et al. found that the user feedback from domains is heterogeneous, and that they should thus consider the data heterogeneity in CST. They extracted the principle coordinates of both users and items and then transferred these coordinates to reduce data sparsity in the target domain. In their experiments, CST can outperform both the non-transfer baselines (e.g., average filling) and the transfer baselines (e.g., CMF).

Centring on active transfer learning, Zhao et al. proposed a framework for CSR in [138]. This approach can be applied in both CDR and CSR scenarios. This study mainly focuses on constructing entity correspondence to transfer useful knowledge across RSs. In each training period, the proposed approach selects a certain number of relevant users whose latent factors are the worst in the target domain. These latent factors are then replaced by the corresponding latent factors in the source domain and, after many iterations, the whole recommendation performance in the target domain is improved eventually. This active transfer learning framework is based on the latent factor model, MMMF. In [139], Zhao et al. extended their framework that was proposed in [138] so that it was based on the other two MF models (RLMF and PMF); they thus posited a unified framework for CSR.

Li et al. suggested a cross-domain framework that improves the recommendation quality in the target domain in [61]. First, they identified the overlaps between users and/or items (common users or items) and then transferred the useful knowledge that was learned from the source domain to help the target domain.

Recently, based on transfer learning, Zhang et al. proposed a series of approaches for single-target CDR [132, 131, 133]. Unlike the traditional single-target CDR approaches, in [132], Zhang et al. proposed a consistent information transfer (CIT) approach which can make sure the knowledge extracted from the source domain is consistent with the target domain. In [131], they further proposed a kernel-induced knowledge transfer (KerKT) approach which employs domain adaptation techniques to transfer knowledge for overlapping entities, and employs a diffusion kernel to correlate the non-overlapping entities. Also, in [133], they proposed a probabilistic knowledge transfer (ProbKT) approach which combines the domain-shared knowledge and the domain-specific knowledge to further improve the recommendation accuracy.

There have recently been many other approaches that are based on transfer learning for single-target CDR [115, 89, 98, 40, 41, 73, 43]. Most tend to transfer useful information or knowledge from a source domain to help a target domain. In [115], Wang et al. transferred the knowledge that was learned from similar mailing lists that could provide useful extra information for broadcast email prioritisation in a target mailing list. In [89], Rafailidis et al. posited a collaborative ranking approach for CDR that uses a weighting strategy to control the importance of latent factors of common users from different auxiliary domains. In [98], Shang et al. proposed a probabilistic MF approach for CDR that leverages the auxiliary ratings and user demographics from the source domain to help the target domain.

Additionally, based on transfer learning, Huang et al. posited a low-rank and s-parse cross-domain (LSCD) approach for CDR that extracts the latent factors of users and items for each respective domain [43]. Hu et al. proposed a transfer-learning framework for CDR called collaborative cross networks (CoNet) in [40]. CoNet uses neural networks for learning complex user-item interaction relations, as well as cross-mappings for connecting two base networks and transferring the knowledge from one to another. In [41], Hu et al. proposed another framework for CDR, called transfer meeting hybrid (TMH), which also considers unstructured text information. This s-tudy involved first extracting useful content information by using a memory network and then selectively transferring knowledge across domains by using a transfer network. In [73], Manotumruksa et al. posited a transfer learning-based CDR approach to make venue recommendations.

2.2.2.3 Clustering-Based Approaches

In addition to transfer learning-based approaches, there are single-target CDR approaches that are based on clustering [90, 88, 22, 119]. In [90], Ren et al. suggested a novel probabilistic cluster-level latent factor (PCLF) framework for CDR that captures the diversities among different domains. In [88], Rafailidis et al. posited a joint cross-domain user clustering and similarity learning (JCSL) framework, which makes cross-domain recommendations by focusing on clustering users and transfer useful knowledge for similar users across domains.

Additionally, Farseev et al. suggested a cross-network collaborative recommendation framework based on multi-view social data called C^3R, which includes both individual and group knowledge [22]. In [119], Wang et al. proposed a cross-domain item embedding framework based on co-clustering (CDIE-C) that considers single-domain and cross-domain sessions.

2.2.2.4 Deep Neural Networks-Based Approaches

Neural networks are widely used in existing CDR approaches [71, 46, 36, 27, 67]. In [71], Man et al. proposed the embedding and mapping framework for CDR (EMCDR). EMCDR applies a linear mapping strategy [77] — linear matrix translation — and a non-linear mapping strategy [105] — MLP — to map the latent factors of common users/items from the source domain and to fit them in the target domain. EMCDR uses the two latent factor models MF [53] and BPR [91] to generate the latent factors of users and items. After the mapping process, the affine latent factors can improve the quality of latent factors in the target domain. The training process aims to minimise the mapping loss from the source domain to the target domain.

Additionally, Jaradat et al. focused on neural networks and first considered the relations between visual and textual inputs; they then transferred useful knowledge from complex domains for the purpose of efficient recommendations [46]. In [36], He et al. posited a CDR framework, based on a Bayesian neural network, that bestowed flexible weights on the latent factors that were learned from different domains. In [27], Fu et al. proposed the review and content-based deep fusion model (RC-DFM) for CDR. Based on four stacked denoising autoencoders models, RC-DFM fuses review texts and item details with the rating matrix. Therefore, the latent factors that are learned in this way remain both ratings and content information. Finally, RC-DFM applies MLP to transfer knowledge across domains. In [67], Liu et al. suggested the deep aesthetic cross-domain network (ACDN), which considers personal aesthetic preferences and shares these preferences across domains to improve recommendation accuracy.

2.2.2.5 Others

There are other single-target CDR approaches that are based on different techniques and ideas. This subsection briefly introduces some of these representative works.

In [60], Li et al. attempted to study a new direction in CDR: cross-domain CF over time. The authors proposed a temporal-domain CF framework that shares the static rating matrix across domains and captures the user-interest drift over time. This proposed method is based on a Bayesian latent factor model called Bi-LDA [87].

To further improve the recommendation performance of existing CDR approaches, Hu et al. [42] posited a generalised cross-domain triadic factorisation (CDTF) model that considers the triadic relation (i.e., the user-item-domain relation). This triadic relation can effectively represent the interactions between domain-specific user latent factors and item latent factors. Additionally, this study leverages both explicit and implicit feedback to address the data sparsity problem.

In [68], Loni et al. suggested a factorisation machines-based approach for CDR. The basic concept of this approach involves first generating the latent factors in each domain using factorisation machines, then leveraging user interaction patterns to transfer knowledge across domains. In the study's experiments, this approach had lower computational complexity and a better recommendation performance when compared to the baselines.

Using a classical reinforcement learning model (i.e., contextual bandit) as a basis, Liu et al. posited a framework for CDR called transferable contextual bandit (TCB) [66]. The contextual bandit policy can gradually exploit and explore user interests. The proposed TCB not only benefits the exploitation process in each domain, but it also improves the exploration process in the target domain. In turn, the training of the exploration process can help inform how the knowledge across domains can be transferred.

As introduced above, MTL, transfer learning, neural networks and reinforcement learning have been widely used in existing CDR approaches. In addition to these techniques, relational learning [103] and semi-supervised learning [49] are also applied in CDR. In [103], Sopchoke et al. employed relational learning to generate the rules that can explain why items were recommended to a target user. In [49], Kang et al. proposed a semi-supervised learning-based CDR framework (SSCDR) that can effectively map the latent factors across domains.

Many of the existing CDR approaches tend to leverage common users as a bridge that links two domains, and they tend to share their latent factors or rating patterns across domains. However, in [28], Gao et al. believed that the personal information and operating information of users in many RSs are not easily shared in public. Therefore, the authors suggested a CDR approach — the neural attentive-transfer recommendation (NATR) approach — that does not share any user information. In NATR, only the items' information is considered, and the item embedding is transferred across domains.

Most of the existing single-target CDR approaches tend to transfer latent factors across domains. In contrast to these approaches are CDR approaches that transfer rating patterns across domains [58, 29, 37, 128]. Li et al. proposed another CDR framework, the codebook-based knowledge transfer (CBT) framework, for movie recommendations [58]. Unlike RMGM, this CBT tends to transfer the rating patterns from a dense auxiliary domain to a sparse target domain (movie domain $\rightarrow$ book domain). CBT does not require user or item overlap (i.e., common users or items). In-

Table 2.8: The comparison of existing MDR approaches

Category	Representative approaches	Training Data	Technology adoption or basic idea
Multi -Domain Recommendation	MCF – Zhang et al. [135]	Ratings	Transfer learning & PMF
	Moreno et al. [80]	Ratings	Feature combination
	TCF – Pan et al. [83, 85]	Binary ratings	Transfer learning
	Zhang et al. [137]	Ratings	Active learning

stead, based on the known ratings in the target domain, it builds a bridge that links the two domains and transfers the user rating pattern from the movie domain to the book domain. The experimental results demonstrate that, compared to the baselines, CBT can address the data sparsity problem by transferring useful knowledge (the rating patterns) and thus improve recommendation performance. In [29], Gao et al. posited a transfer learning-based framework that transfers rating patterns across domains. This work pools the known ratings from different domains together and shares the latent rating pattern across domains. Based on the posited cluster level-based latent factor model, the approach not only shares the rating pattern across domains, but it also learns the domain-specific rating patterns of users in each domain for improving recommendation accuracy. In [37], He et al. also proposed a novel CDR approach for transferring rating patterns across domains, while Yuan et al. proposed a deep domain adaptation model (called DARec) for transferring rating patterns across domains [128]. Yuan et al. extracted the rating patterns from rating matrices, with the rating patterns being independent of the auxiliary information from the source domain.

2.2.3 Multi-Domain Recommendation

MDR is another highly relevant direction to CDR, but it achieves a different goal: it makes recommendations for all domains. Some of these multi-domain approaches can be applied in CDR scenarios, but they either tend to make recommendations for specific or common users who are selected from domains, or only for the users in the target domain. For a clear comparison, these MDR approaches are detailed in Table 2.8. This section also introduces relevant MDR approaches below.

In [135], Zhang et al. posited a multi-domain collaborative filtering (MCF) framework for solving the data sparsity problem in multiple domains. The authors used PMF to generate the latent factors that were learned from different domains, as well as to adaptively transfer useful latent factors across domains. They also used a link function to correct the domains' biases.

Moreno et al. suggested a multi-domain framework that leveraged the auxiliary information from multiple domains to help a sparse target domain [80]. This approach does not require user overlap (common users) to link the auxiliary multiple domains with the target domain. It integrates useful knowledge from different domains to improve the recommendation accuracy in the target domain.

Based on transfer learning, Pan et al. proposed a multi-domain framework called transfer by collective factorisation (TCF) — which considers heterogeneous user feedback — to predict unknown ratings in [83, 85]. In these studies, Pan et al. found that the RSs may lack richer data in numerical ratings because users often provide binary ratings (e.g., like and dislike). The authors focused on transferring the binary ratings to a target numerical rating matrix, and thus their TCF can capture the data-dependent effect and improve the quality of knowledge transfer.

Recently, in [137], Zhang et al. proposed an active learning strategy that addresses the data sparsity problem in multi-domain scenarios. The authors considered both domain-specific and domain-independent knowledge. Then, to measure the generalisation error of domain-specific and domain-independent knowledge, they used an expected entropy and a variance-based strategy, respectively. Finally, the proposed approach applied an active learning strategy that combined the two measurements and shared this global knowledge across domains.

2.2.4 Single-Target CDR: A Summary

As discussed above, the single-domain recommendation approaches are constrained by limited data, and existing single-target CDR approaches aim to address the data sparsity problem on a single domain. Compared to SDR approaches, single-target CDR approaches can leverage richer information from the source domain to improve the recommendation accuracy in the target domain. The existing single-target CDR approaches tend to use two main strategies: leveraging content information to link two domains and then share the useful information across domains (content-based transfer), or leveraging some latent factor models or neural networks to generate the latent factors of users in each domain and then share these latent factors across domains (feature-based transfer).

Content-based transfer approaches tend to leverage multi-source or multi-view content information as a bridge for linking two domains (e.g., user/item attributes, social tags, semantic properties, thumbs-up, metadata and browsing or watching history). To extract useful content information, these approaches can employ topic models that generate latent topics of users/items, or semantic analysis models that generate semantic features of users/items. Based on the content similarities, these approaches can discover similar users/items and then transfer useful user preferences or item details across domains.

Feature-based transfer approaches also tend to employ machine learning techniques to achieve single-target CDR, such as MTL, transfer learning, clustering, reinforcement learning, deep neural networks, relational learning and semi-supervised learning. These approaches first generate latent factors or rating patterns in each domain and then directly transfer or map these useful features across domains.

In addition to these single-target CDR approaches, MDR approaches tend to achieve a different but relevant goal: making recommendations by leveraging the auxiliary information from multiple domains to help a target domain. These MDR approaches tend to employ transfer learning, active learning or feature combination to share useful information among different domains.

Although CDR has been studied from several angles, it is still a challenging and under-explored topic in RSs [92]. Most CDR approaches are single-target approaches, which means that they cannot leverage any information in the target domain to assist the source domain. The researchers observed that the target domain may be richer in certain types of data even if it may be very sparse in many types of data. This signifies that the information in the target domain could be used to improve the source domain if this information is used effectively. In fact, this is the basic motivation for proposing new dual-target CDR approaches in the following chapters.

2.3 Dual-Target Cross-Domain Recommendation

Dual-target CDR is still a novel concept for improving the recommendation accuracies in both domains simultaneously. Therefore, existing solutions are limited. The existing dual-target CDR approaches mainly focus on applying fixed or flexible combination strategies [141, 143], or they focus on simply changing the existing single-target transfer learning to become dual-transfer learning [63].

In [141], Zhu et al. first proposed the DTCDR, a dual-target CDR framework that uses multi-source information such as ratings, reviews, user profiles, item details and tags to generate more representative embeddings of users and items. Then, based on MTL, the DTCDR framework uses three different combination strategies to combine and share the embedding of common users across domains.

Additionally, in [63], Li et al. posited the DDTCDR, a deep dual-transfer framework for dual-target CDR. The DDTCDR framework considers the bidirectional latent relations between users and items and applies a latent orthogonal mapping to extract user preferences. Based on the orthogonal mapping, DDTCDR can transfer users' embeddings in a bidirectional way (i.e., Source $\rightarrow$ Target and Target $\rightarrow$ Source).

Recently, Zhu et al. proposed another dual-target CDR framework in [143] — the GA-DTCDR. The authors first constructed separate heterogeneous graphs for each domain to generate more representative embeddings of users and items. They then proposed an element-wise attention mechanism to effectively combine the embeddings from both domains. Using this method, they could then improve the recommendation accuracy in both domains.

2.3.1 Dual-Target CDR: A Summary

Dual-target CDR is still a new direction in the field of CDR, one that aims to improve the recommendation accuracy in both domains simultaneously. To achieve dual-target CDR, it is necessary to effectively utilise data richness and diversity in different domains. Existing dual-target CDR approaches either attempt to combine and share the embedding of common users or items across domains, or they extend the classical single-target CDR approaches to be applied in dual-target CDR scenarios.

2.4 Multi-Task Learning

In Chapter 4, the researchers proposed a dual-target CDR approach that is based on MTL. Therefore, this section will review the most relevant works in the area of MTL. According to the correlations with RSs, these MTL approaches are classified into two main categories — MTL approaches and MTL-based recommendation approaches.

- **MTL approaches.** MTL approaches can be broadly summarised by five categories, according to [136] — feature learning, low rank, task clustering, task relation learning and decomposition. Feature-learning approaches [13, 113, 134] can improve the performance of all tasks by sharing a common feature representation that is learned from the data in all tasks. Low rank approaches [129, 2, 34] assume that the parameters of different tasks share a low-rank subspace that can render the parameters of all tasks more accurate. Task-clustering approaches [112, 124, 64] first cluster similar tasks and then share their parameters. The fourth category is based on relational knowledge transfer [75, 76] and the last category decomposes the parameter matrix into two or more component matrices [44, 15, 32], which can eliminate unimportant features and share important features for all tasks.

- **MTL-based recommendation approaches.** The approach proposed in [3] combines correlated context information from multiple tasks to improve predictive accuracy in RSs. Additionally, the multi-task recommendation model proposed in [69] focuses on combining MF for rating prediction and adversarial sequence learning for recommendation explanation.

For a clear comparison, these MTL approaches are detailed in Table 2.9.

2.4.1 MTL Approaches

MTL approaches can be broadly summarised by five categories, according to [136]: feature learning, low rank, task clustering, task relation learning and decomposition. This thesis will only review some of the most relevant works.

2.4.1.1 Feature Learning

First, Rich Caruana proposed an MTL framework in [13] based on k-nearest neighbour, kernel regression and decision trees. In this study, Caruana explained the mechanism of MTL and applied the study's approach in some real domains.

Then, in [113], Titsias et al. suggested a variational Bayesian inference approach for multiple sparse linear models. Based on the spike and slab prior knowledge, the suggested approach can combine the task-specific weights from different tasks and then share such combined weights across tasks according to the relations between two tasks.

Zhang et al. recently proposed a framework for biological image analysis based on transfer learning and MTL in [134]. The authors employed deep convolutional neural networks to act on image pixels directly. In the experiments, they leveraged labelled

Table 2.9: The comparison of existing MTL approaches

Category	Representative approaches	Technology adoption or basic idea
Feature Learning	Caruana et al. [13]	K-nearest neighbor & kernel regression & decision trees
	Titsias et al. [113]	Variational Bayesian inference
	Zhang et al. [134]	Transfer learning & multi-task learning
Low-Rank	Zhang et al. [129]	Independent component
	Agarwal et al. [2]	Manifold regularisation
	RAMUSA – Han et al. [34]	Capped trace norm regulariser
Task Clustering	TC – Thrun et al. [112]	TC clustering
	MSBP – Xue et al. [124]	Matrix stick-breaking
	GP – Lian et al. [64]	Sparse construction & Variational Bayes inference
Task Relation Learning	Mihalkova et al. [75]	Transfer learning & markov logic networks
	SR2LR – Mihalkova et al. [76]	Transfer learning
Decomposition	Jalali et al. [44]	Transfer learning & parameter overlap
	Chen et al. [15]	Cardinality regularisation term & low-rank constraint
	rMTFL – Gong et al. [32]	Accelerated gradient descent

natural images to train the proposed model and extract the generic features. Then, they transferred useful features from the source domain to the target domain.

2.4.1.2 Low-Rank

Based on an independent component strategy, Zhang et al. proposed a probabilistic approach for MTL in [129]. First, the authors considered the relatedness between different tasks to generate the task parameters in each task. Then, to identify the hidden and independent components, they employed Laplace distributions to model the hidden sources. This proposed approach is also compatible with empirical Bayes methods and point estimation.

Then, Agarwal et al. proposed an MTL framework based on manifold regularisation in [2]. This approach assumed that the parameters of all related learning tasks lie on a common manifold. In the training process, the proposed approach learned the task parameters and the corresponding manifold alternately. Based on conventional single-task learning strategies, such as SVMs, the proposed approach learns all task parameters and uses them to learn the corresponding manifold.

Additionally, in [34], Han et al. discovered that a global penalisation on the singular values of the weight matrix may lead to a serious problem — an estimation loss when the proposed approach wants to recover the larger singular values. To address this problem, the authors proposed a Reduced rAnk MUlti-Stage multi-tAsk learning (RAMUSA) approach. Specifically, RAMUSA employs a capped trace norm regulariser to minimise the singular values of the weight matrix only if the singular values are smaller than a certain threshold. In this way, the estimation loss at each stage in RAMUSA shrinks and finally achieves a lower upper-bound.

2.4.1.3 Task Clustering

Thrun et al. [112] posited a task-clustering (TC) approach for MTL. First, TC groups different learning tasks into different classes. Then, for a new learning task, TC can provide useful information from the most relevant task cluster. Finally, such useful information can improve the new learning task.

Further, Xue et al. proposed an MTL framework that was based on the matrix stick-breaking process (MSBP) [124]. The process involves clustering information from different tasks into different feature components. For each task, MSBP tends to share the most related feature component from other tasks to this respective task.

Additionally, Lian et al. posited a multi-task point process model by using a hierarchical Gaussian process (GP) [64]. Specifically, the GP can map historical events to future ratings. Lian et al. also applied a sparse construction in their model and designed a variational Bayes strategy for model learning and inference.

2.4.1.4 Task Relation Learning

Based on transfer learning with Markov logic networks, Mihalkova et al. proposed a framework for transferring the predications from a source domain to a target domain [75]. These transferred predictions can ultimately improve the accuracy in the target domain.

Further to this, in [76], Mihalkova et al. introduced a single entity-centred setting that is an important framework for transferring useful information across domains. In the target domain, this framework only provides information concerning a single entity. Mihalkova et al. posited an SR2LR algorithm that could map the information from the source domain to the target domain.

2.4.1.5 Decomposition

In [44], Jalali et al. proposed an MTL framework for multiple linear regression tasks. The authors leveraged parameter overlapping to transfer useful parameters across tasks. For two related tasks, their proposed dirty model estimated overlap of two sets of parameters and trained them differently.

Further to this, Chen et al. proposed an MTL framework to share the sparse and low-rank patterns rather than the parameters across different related tasks [15]. Based on a linear MTL formulation, the sparse and low-rank patterns were learned by a cardinality regularisation term and a low-rank constraint, respectively. Chen et al. further proposed two projected gradient algorithms to solve the optimisation formulation problems. Apart from these points, the authors also applied presented projected gradient algorithms for MTL formulation.

Additionally, Gong et al. proposed a robust multi-task feature learning (rMTFL) approach that could share a set of common features across tasks [32]. This proposed rMTFL approach can identify outlier tasks (sparsity patterns) by using a group lasso penalty on the column groups of the second component of the weight matrix. The authors employed an accelerated gradient descent for the optimisation of the proposed rMTFL. Their experiments indicated that the proposed rMTFL can capture and share the shared features across tasks, and that it can identify outlier tasks.

2.4.2 MTL-Based Recommendation Approaches

In [3], Agarwal et al. proposed an approach that combined the multiple context information from different tasks to improve the predictive accuracy in RSs. To reduce the bias in estimates, the authors leveraged the positing of local information and context-specific factors of entities. Additionally, to avoid the over-fitting problem caused by data sparsity, the local factors were connected to a shared global structure. Based on the concept of the EM framework, Agarwal et al. eventually trained their proposed model.

Further to this, in [69], Lu et al. proposed a multi-task approach for combining MF for rating prediction and adversarial sequence learning for recommendation explanation.

2.4.3 Multi-Task Learning: A Summary

Existing MTL approaches are classified into five groups: feature learning, low rank, task clustering, task relation learning and decomposition. Most approaches focus on sharing features across tasks, which is an effective method for addressing the data sparsity problem.

In particular, the feature-learning approaches tend to apply classical machine learning techniques — such as k-nearest neighbour, kernel regression, decision trees, variational Bayesian inference, transfer learning and MTL — to generate features or share these features across multiple tasks. The low-rank approaches tend to apply some regularisers (e.g., the manifold regulariser and capped trace norm regulariser) to share a low-rank subspace for different tasks. The TC approaches tend to make clustering strategies, such as TC clustering, to group similar tasks and then share their parameters. The task relation-learning approaches tend to apply transfer learning to the process of transferring useful relations across tasks. The last category involves decomposing the parameter matrix into two or more component matrices by using decomposition techniques.

Although MTL has been applied for RSs, efficiently applying the existing MTL approaches for dual-target CDR is difficult because they heavily rely on side information (i.e., additional information that is associated with the users and items) from a single domain. However, a sparser domain in dual-target CDR may be too sparse to support it.

2.5 Graph Embedding

As Chapter 5 has proposed the novel GA-DTCDR framework for consideration, this section will introduce the most relevant works to the area of graph embedding.

Graph embedding involves learning a mapping function in which the nodes in a graph are mapped to low-dimensional latent representations [140]. These latent representations can be used as the features of nodes for different tasks, such as classification and link prediction. According to embedding techniques, this section classifies the existing graph-embedding approaches into two categories: dimensionality reduction and neural networks. Dimensionality reduction-based approaches — such as multidimensional scaling [54], principal component analysis [121] and their extensions [125] — involve optimising a linear or non-linear function that reduces the dimension of a graph's representative data matrix and then produces low-dimensional embeddings. Neural network-based approaches — such as DeepWalk [86], LINE [109] and Node2vec [33] — involve treating nodes as words and the generated random walks on graphs as sentences, and then learning node embeddings based on these words and sentences [140].

For a clear comparison, these graph embedding approaches are listed in detail in

Table 2.10: The comparison of existing graph embedding approaches

Category	Approaches	Technology adoption or basic idea
Dimensionality reduction-based	MDS – Kruskal et al. [54]	Dimensionality reduction
	PCA – Wold et al. [121]	Dimensionality reduction
	MFA – Yan et al. [125]	Linear & non-linear mappings
	Shi et al. [100]	Feature combination
Neural network -based	DeepWalk – Perozzi et al. [86]	Random walks
	LINE – Tang et al. [109]	Local and global structures & edge-sampling strategy
	Node2vec – Grover et al. [33]	Neighbourhood-based
	Cao et al. [12]	Random surfing strategy
	SDNE – Wang et al. [117]	Semi-supervised deep learning
	Kipf et al. [51]	Variational auto-encoder
	Pan et al. [81]	Topological structure & node content
	Tu et al. [114]	Regular equivalence
	NetRA – Yu et al. [127]	Locality-preserving & global reconstruction constraints

Table 2.10.

2.5.1 Dimensionality Reduction-Based Approaches

To uncover the hidden structure of databases, Kruskal et al. proposed a spatial representation approach — multidimensional scaling (MDS) — for a dataset in [54]. MDS is a conventional approach that maps a data graph to a representative matrix. The general concept of this approach is to optimise a linear function to reduce the dimension of the representative matrix.

In [121], Wold et al. posited a basic method called principal component analysis

(PCA) for analysing the multivariate data, and it can capture essential data patterns. The basic concept of this method involves achieving an approximation of a data table or data matrix. PCA can be used to generate the low-dimensional latent representations of a data matrix (i.e., graph embedding).

In addition to the above two conventional approaches, there are also many dimensionality reduction-based approaches that focus on other aspects, such as marginal fisher analysis (MFA) [125] and graph joint graph embedding and sparse regression [100]. In [125], Yan et al. proposed a general framework for graph embedding that contains a linear function and a non-linear kernel for mapping a data graph to low-dimensional embeddings. In [100], Shi et al. proposed a unified framework for all supervised, semi-supervised and unsupervised learning tasks. In this approach, Shi et al. tended to combine the objective functions of graph embedding and sparse regression, which can make graph embedding and sparse regression implement and optimise simultaneously.

2.5.2 Neural Network-Based Approaches

In [86], Perozzi et al. proposed the well-known graph-embedding approach, DeepWalk, which learns the latent representations of nodes in a graph. In DeepWalk, social relations are encoded in the latent representations. DeepWalk is applied in language-modelling tasks and deep-learning tasks in [86], which achieves an effective performance. Specifically, DeepWalk can learn the latent representations by treating random walks as the equivalent of sentences. Additionally, based on online learning, DeepWalk can also be considered a scalable approach.

Tang et al. focused on embedding large information graphs into low-dimensional vector spaces in [109], and they proposed an effective embedding approach — LINE — for many learning tasks (e.g., node classification and link prediction). LINE is suitable for different types of information graphs, such as undirected graphs, directed graphs and weighted graphs. Specifically, regarding their objective function, they consider both local and global network structures. Additionally, to address the problems caused by the classical stochastic gradient, they also take an edge-sampling strategy.

Grover et al. suggested a graph-embedding framework — Node2vec — that could generate continuous feature representations for nodes in a graph [33]. In this study, the authors believed that exploring the neighbourhoods of nodes in a graph is the key to learning richer representations. Based on this belief, they mapped the nodes in a graph to low-dimensional vector spaces.

Further to this, there have recently been many graph-embedding approaches that are based on DNNs [12, 117, 51, 81, 114, 127]. Unlike traditional graph-embedding s-trategies, Cao et al. [12] employed a random surfing strategy to directly capture graph structural information, rather than using some sampling-based strategies to generate linear sequences. After this, Wang et al. [117] proposed a structural deep network embedding approach (SDNE) to capture a complex and non-linear network structure. SDNE employs a semi-supervised deep model for non-linear network structure and jointly optimises first-order and second-order proximity for preserving both local and global network structures. Based on a variational auto-encoder, Kipf et al. proposed a graph-embedding approach for undirected graphs [51]. Additionally, Pan et al. proposed an adversarial graph-embedding framework in [81]. In this study, Pan et al. considered both topological structure and node content when they encoded the nodes in a graph. Tu et al. proposed a deep recursive network embedding (DRNE) approach in [114]. Using the DRNE approach, Tu et al. considered regular equivalence (i.e., that many nodes in different parts of a graph may have similar roles) when they wanted to map the nodes in the graph to low-dimensional vectors. Finally, to address the sparsity problem in graph embedding, Yu et al. aimed to learn the network representations with adversarially regularised auto-encoders (NetRA) in [127]. NetRA considers both locality-preserving and global reconstruction constraints to capture the graph's structure.

2.5.3 Graph Embedding: A Summary

Classical graph-embedding approaches tend to use dimensionality reduction-based strategies to embed the nodes in a graph, while recent neural network-based approaches tend to employ DNNs to learning a non-linear representation of nodes. Existing dimensionality reduction-based approaches focus on reducing the dimensionality of representation, on learning linear/non-linear mappings or on combining features. Existing neural network-based approaches tend to consider the structure of a graph and then employ machine learning strategies or techniques (e.g., random walks, edge-sampling strategies, neighbourhood-based algorithms and semi-supervised learning) to learn the embeddings of nodes.

Graph embedding is a key tool for many learning tasks, such as visualisation, node classification and link prediction, which can provide the latent representation of nodes in a graph in the form of vectors. In RSs, the different types of relations — such as user-user, item-item and user-item relations — can be included as a relation graph. The graph-embedding technique can thus play an important role in RSs. In fact, Chapter 5 outlines how this technique will be employed as an important tool for generating the rating embeddings of users and items.

2.6 Attention Mechanism

As Chapter 5 proposed the GA-DTCDR framework, this section will briefly introduce the most relevant approaches to the attention mechanism.

Attention is first introduced in [5], which involves providing a more accurate alignment for each position in a machine translation task. Apart from machine translation, the attention mechanism has also recently been used in image captioning [126, 17] and recommendation [16]. The general concept of the attention mechanism is to focus on selective parts of the whole information, which can capture the outstanding features of objects. For recommendation, the existing attention approaches [16, 39, 111, 118] tend

Table 2.11: The comparison of existing attention-based approaches

<table>
<tr><th>Application field</th><th>Approaches</th><th>Technology adoption
or basic idea</th></tr>
<tr><td>Machine translation</td><td>Bahdanau et al. [5]</td><td>Encoder-decoder</td></tr>
<tr><td rowspan="2">Image captioning</td><td>You et al. [126]</td><td>Semantic concept proposals</td></tr>
<tr><td>SCA-CNN –
Chen et al. [17]</td><td>Attentive spatial locations
& attentive channels</td></tr>
<tr><td rowspan="4">Recommendation</td><td>ACF –
Chen et al. [16]</td><td>item-level and
component-level attentions</td></tr>
<tr><td>Hu et al. [39]</td><td>Meta-path based context</td></tr>
<tr><td>MPCN –
Tay et al. [111]</td><td>Importance of reviews</td></tr>
<tr><td>KGAT –
Wang et al. [118]</td><td>Neighbors-based
embedding propagation</td></tr>
</table>

to select information parts of explicit or implicit data to improve the representations for users and items.

In [5], Bahdanau et al. first proposed an attention framework for English-to-French translation. They extended the traditional encoder-decoder methods and automatically linked a source sentence part to a target word. Especially in the decoder process, Bahdanau et al. implemented an attention mechanism to pay attention to parts of the source sentence.

The attention mechanism is also widely used in image captioning [126, 17]. For example, in [126], You et al. proposed a semantic attention framework that combines the conventional top-down and bottom-up approaches in the field of image captioning. This approach can selectively pay attention to semantic concept proposals and map these proposals onto latent spaces. Using convolutional neural networks as a basis, Chen et al. [17] proposed a spatial and channel-wise attention mechanism — SCA-CNN — for image captioning. Specifically, SCA-CNN can dynamically generate sentences and encode attentive spatial locations at multiple layers and attentive channels.

Recently, the attention mechanism is also applied in RSs. This thesis will only introduce a part of these approaches so that their research directions and basic concepts can be briefly understood. To achieve multimedia recommendation, Chen et al. [16] proposed a novel attention mechanism in CF — ACF — to leverage item-level and component-level historical feedback. In [39], and using a co-attention mechanism as a basis, Hu et al. proposed a DNN for top-N recommendations. In this approach, the co-attention mechanism can improve the representations of users and items by leveraging meta-path-based contexts. Tay et al. believed that the importance of different reviews varies to some extent [111]. The authors thus proposed a multi-pointer co-attention network (MPCN) that can extract important reviews from users and items. Additionally, in [118], Wang et al. proposed a knowledge graph attention network (KGAT) for making recommendations, which considers the utility of knowledge graphs and links items with their attributes. The basic concept of KGAT is to propagate the embeddings from a node's neighbours in a graph to the embedding of the target node. During the propagation process, the attention mechanism is used to measure the importance of the neighbours.

2.6.1 The Attention Mechanism: A Summary

The attention mechanism has been widely used in many application fields, such as in machine translation, image captioning and recommendation. Its general concept involves paying more attention to the important parts of the target data and offering them higher weights in the neural networks. In the field of machine translation, existing attention approaches directly apply encoder-decoder techniques to translate one language to another. In the field of image captioning, with the help of the attention mechanism and other techniques (e.g., semantic analysis), existing approaches can provide accurate captions to images. Additionally, in the field of recommendation, the attention mechanism can help RSs offer high weights to important information (e.g., reviews, item details and metadata).

Since the attention mechanism can automatically provide the weights to the embeddings of users and items from different data sources, it can thus be employed to improve the quality of embedding combinations in dual-target CDR. This is why it was chosen as the combination technique in Chapter 5.

2.7 Summary

In this chapter, the researchers reviewed the related works pertaining to SDR, single-target CDR and dual-target CDR.

In regard to existing SDR approaches, their main categories, characteristics, advantage and disadvantage, can be summarised as follows:

- **Categories and characteristics.**
  - ✓ **Rating-based approaches.** These approaches tend to leverage observed ratings to predict unknown ratings. The training process involves minimising the rating loss or rating ranking loss. However, the rating matrices in these RSs are sparse.
  - ✓ **Content-based approaches.** These approaches tend to leverage observed ratings and content information to predict unknown ratings. However, the content information is still limited to a single domain.
- **Advantage.** These SDR approaches can partly address the data sparsity problem in a single domain by taking some effective strategies, e.g., collaborative filtering, and leveraging multi-source information, e.g., ratings and contents.
- **Disadvantage.** These SDR approaches are constrained by limited data from a single domain, which signifies that using these approaches to solve the data sparsity problem will be difficult.

In regard to existing single-target CDR approaches, their main categories, characteristics, advantage and disadvantage, can be summarised as follows:

- **Categories and characteristics.**

  ✓ **Content-based transfer.** These approaches first leverage content information as a bridge that links different domains, and then they transfer user preferences or item details across domains. However, the rating matrices in these RSs are sparse.

  ✓ **Feature-based transfer.** These approaches first generate latent factors of users/items or rating patterns in each domain, and then they transfer these features across domains.

- **Advantage.** These CDR approaches can leverage the auxiliary information from the source domain to improve the recommendation performance in the target domain.

- **Disadvantage.** These CDR approaches are single-target approaches, which means that they cannot leverage any information in the target domain to help the source domain.

In regard to existing dual-target CDR approaches, they mainly focus on applying fixed combination strategies or on extending the classical single-target CDR approaches so they can be applied in dual-target CDR scenarios. Dual-target CDR is still a novel direction, but it is a promising direction that could be used to further improve the recommendation performance of existing single-target CDR approaches.

Additionally, in Chapter 4, MTL was used as a basis for proposing a dual-target CDR approach. The chapter thus also briefly introduced the most relevant approaches pertaining to MTL. For existing MTL approaches, their main categories, characteristics, advantage and challenge, can be summarised as follows:

- **Categories and characteristics.**

  ✓ **MTL approaches.** These approaches generally use five different strategies to share useful data or features across tasks: feature learning, low rank, task clustering, task relation learning and decomposition.

  ✓ **For recommendation tasks.** These approaches tend to combine multi-source information or features from different tasks to improve the predictive accuracy in RSs.

- **Advantage.** These MTL approaches can share a part of parameters or knowledge learned from different tasks to improve the performance in a target task.

- **Challenge.** Efficiently applying these MTL approaches for dual-target CDR is difficult because they heavily rely on side information or on other additional information.

Similarly, Chapter 5 used the graph-embedding technique and attention mechanism as a basis for proposing the GA-DTCDR. The most relevant approaches pertaining to these two fields were thus briefly reviewed in the last part of this chapter.

In regard to existing graph-embedding approaches, their main categories, characteristics, advantage and challenge, can be summarised as follows:

- **Categories and characteristics.**

  ✓ **Dimensionality reduction-based approaches.** These approaches use dimensionality-reduction strategies for learning linear/non-linear mappings or for combining features.

  ✓ **Neural network-based approaches.** These approaches consider a graph's structure and employ different machine learning techniques for learning the embeddings of nodes.

- **Advantage.** For RSs, these graph-embedding approaches has the potential to generate more representative embeddings of users and items.

- **Challenge.** It is still a challenge for these graph-embedding approaches to construct an informative graph.

In regard to existing attention mechanism approaches, their main categories, characteristics, advantage and challenge, can be summarised as follows:

- **Categories and characteristics.**

  ✓ **Machine translation approaches.** These approaches tend to apply encoder-decoder techniques to translate one language to another.

  ✓ **Image captioning approaches.** These approaches tend to consider additional information, such as semantic information, to provide accurate captions for images.

  ✓ **Recommendation approaches.** These approaches tend to attribute high weights to important information, such as reviews, item details and metadata.

- **Advantage.** To target dual-target CDR problem, these attention mechanism approaches can automatically give appropriate weights to the embeddings of common users or common items from both domains.

- **Challenge.** It is still a challenge for these attention mechanism approaches to design an appropriate activation function and a reasonable loss function for attention networks.

Chapter 3

A Deep framework for Cross-Domain and Cross-System Recommendations

To address the data sparsity problem in RSs, a new trend has emerged in recent years that utilises relatively richer information (e.g., ratings) from the source domain or system to improve the recommendation accuracy in the target domain or system. This new trend can be classified as CDR [8] and CSR [138].

Existing transfer-based approaches in CDR and CSR either directly replace a part of the latent factors of users/items from the source domain with the corresponding latent factors in the target domain [139] (Category 1), or they map the latent factors of common users/items in the source domain to fit those factors in the target domain [71] (Category 2). However, the approaches in Category 1 ignore the complex relationship between the latent factors in the two domains, while the approaches in Category 2 only focus on the common users and items so that their relatively accurate latent factors in the source domain or system can be adjusted to fit the worser ones in the target domain, which is neither reasonable nor effective. Therefore, to further improve recommendation accuracy, it is crucial to find an effective method for accurately mapping latent factors across domains or systems. This has introduced in Section 1.2.1 and expressed by **CH1**: *'How can an accurate mapping of the latent factors across domains be found for enhancing recommendation accuracy?'*.

In contrast to existing CDR and CSR approaches, this chapter will propose a novel approach to generating benchmark factors that combines the features of the latent factors in both the source and target domains. The latent factors in the target domain or system are then mapped to fit the benchmark factors. To the best of the researchers' knowledge, this leads to a new category of transfer-based approaches for mapping latent factors across domains or systems, and this thesis's approach is the first one in this novel category.

This section will first formulate the CDR and CSR problems. The DCDCSR will then be proposed and the DNN mapping process for mapping latent factors across domains or systems will be introduced. This section will also describe how cross-domain and cross-system recommendations are made based on predicted ratings. The detailed framework is explained in the following sections.

3.1 Notations and Problem Definition

Let $\boldsymbol{R}^s$ and $\boldsymbol{R}^t$ denote the rating matrices of the source and target domains or systems, respectively. Let $\mathcal{U} = \{u_1, ..., u_n\}$ and $\mathcal{V} = \{v_1, ..., v_m\}$ denote the user and item sets, respectively, where n is the number of users and m is the number of items. $r^t_{ij} \in \boldsymbol{R}^t$ denotes the rating that u_i gives to an item v_j in the target domain or system. Given a rating matrix of $\boldsymbol{R}$, after MF, $\boldsymbol{R}$ is factorized into two latent matrices $\boldsymbol{U}$ $(K \times n)$ and $\boldsymbol{V}$ $(K \times m)$, where K is the dimension of factors. $\boldsymbol{U}$ and $\boldsymbol{V}$ represent the low-rank factor matrices for $\mathcal{U}$ and $\mathcal{V}$, respectively. Concretely, $\boldsymbol{U}^t_i$ denotes u_i's latent factor vector in the target domain or system. Based on these notations, the CDR problem can be defined as below.

Definition 1. CDR problem:

- **Input:** Two observed domains including the rating matrices $\boldsymbol{R}^s$ and $\boldsymbol{R}^t$, the user sets $\mathcal{U}^s$,$\mathcal{U}^t \subseteq \mathcal{U}$, and the item sets $\mathcal{V}^s, \mathcal{V}^t \subseteq \mathcal{V}$.

- **Output:** Recommend the items $\mathcal{V}_i \subseteq \mathcal{V}^t$ to a target user $u_i \in \mathcal{U}^t$ by utilising both $\boldsymbol{R}^s$ and $\boldsymbol{R}^t$.

Similarly, the CSR problem can also be formulated by replacing 'domain' with 'system' in Definition 1.

3.2 The DCDCSR Framework

To target the above problems for both CDR and CSR, a deep framework called the DCDCSR was proposed. This framework can be divided into three phases: **Phase 1**: MF modelling, **Phase 2**: DNN mapping and **Phase 3**: Cross-domain and cross-system recommendations. The framework structure is illustrated in Figure 3.1.

In Phase 1, the user and item latent factor matrices $\{\boldsymbol{U}^s, \boldsymbol{U}^t, \boldsymbol{V}^s, \boldsymbol{V}^t\}$ are obtained by using MF. In Phase 2, the benchmark factor matrices $\{\boldsymbol{U}^b, \boldsymbol{V}^b\}$ are first generated by combining the latent factor matrices $\{\boldsymbol{U}^s, \boldsymbol{U}^t, \boldsymbol{V}^s, \boldsymbol{V}^t\}$ according to the sparsity degrees of individual users and items. Then, the deep neural network in the *Feedforward* and the *Backpropagation* processes is trained to map the latent factor matrices $\{\boldsymbol{U}^t, \boldsymbol{V}^t\}$ to fit the benchmark factor matrices $\{\boldsymbol{U}^b, \boldsymbol{V}^b\}$. In Phase 3, based on the affine factor matrices $\{\hat{\boldsymbol{U}}^t, \hat{\boldsymbol{V}}^t\}$ that were learned from Phase 2, the users' ratings on all items in the target domain or system are predicted, and matched items are recommended to target users. The three phases in the framework are presented in Algorithm 1, with details explained in the following sections.

3.3 Phase 1: MF Modeling

To study the generalisability of the proposed DCDCSR framework in Phase 1, two classical rating-oriented MF models are applied (MMMF [104] and PMF [78]), as well as a representative ranking-oriented MF model (BPR [91]) to generate user and item latent factors for the following mapping process. While the rating-oriented MF models focus on minimising the error between observed and predicted ratings, the ranking-oriented MF model emphasises the personalised rating rankings on items remaining unchanged between observed and predicted ratings — both of which can carry

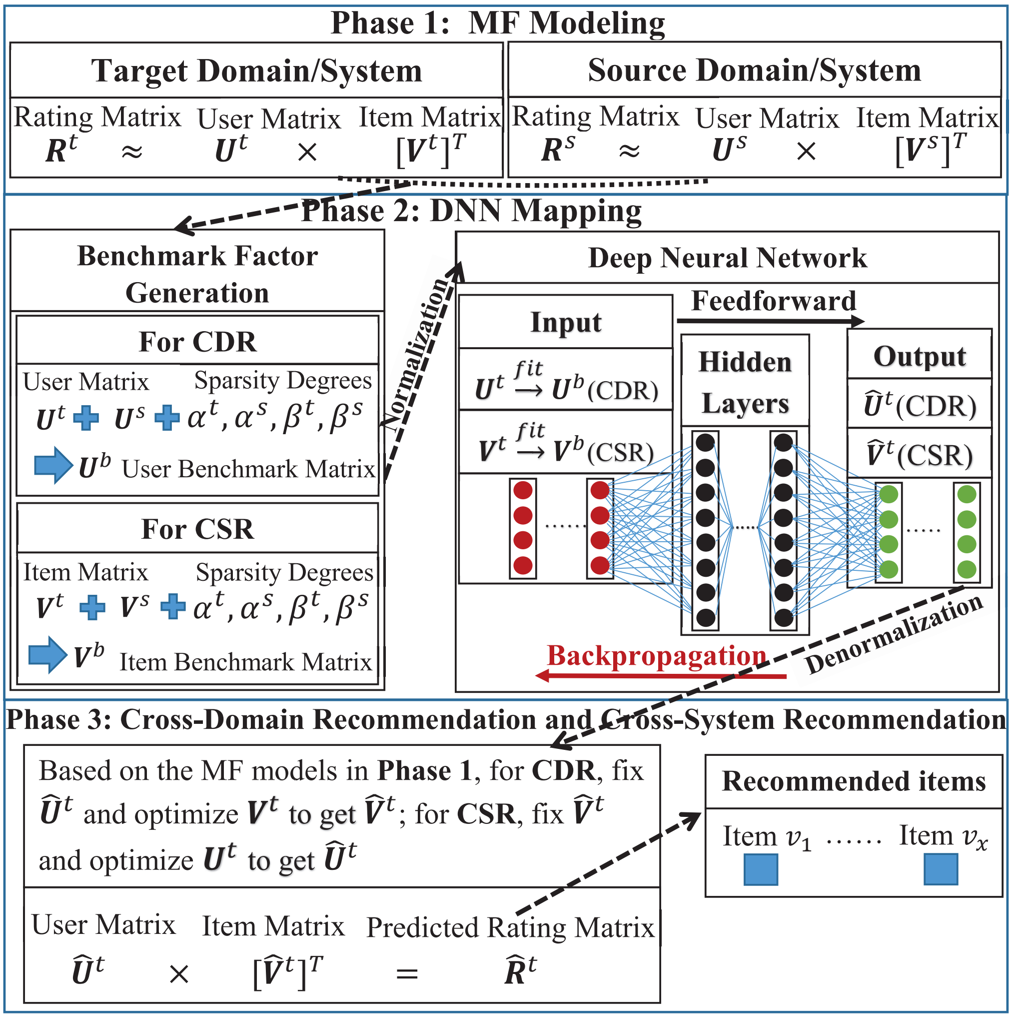


Figure 3.1: The structure of our DCDCSR Framework

different biased latent factors to the following DNN mapping process. The details of the MF models are described in the following subsections.

3.3.1 Rating-Oriented Matrix Factorisation

3.3.1.1 Maximum-Margin Matrix Factorisation

MMMF involves learning a matrix $\hat{R}$ to fit the observed rating matrix R by minimising a trace norm of R and by maximising the corresponding predictive margin. The

Algorithm 1 The DCDCSR Framework

Require: The rating matrices, user sets, and item sets of the source and target domains or systems $\boldsymbol{R}^s$, $\boldsymbol{R}^t, \mathcal{U}^s, \mathcal{U}^t, \mathcal{V}^s$ and $\mathcal{V}^t$.

Ensure: Recommend items $\mathcal{V}_i \subseteq \mathcal{V}^t$ to a target user u_i in the target domain or system.

Phase 1: MF Modelling

Learn $\{\boldsymbol{U}^s, \boldsymbol{V}^s\}$ from $\boldsymbol{R}^s$ by using MF;

Learn $\{\boldsymbol{U}^t, \boldsymbol{V}^t\}$ from $\boldsymbol{R}^t$ by using MF.

Phase 2: DNN Mapping

Generate the benchmark factor matrix $\boldsymbol{U}^b$ for CDR or $\boldsymbol{V}^b$ for CSR.

Normalise $\{\boldsymbol{U}^t, \boldsymbol{U}^b\}$ for CDR or $\{\boldsymbol{V}^t, \boldsymbol{V}^b\}$ for CSR.

Train the parameters of the deep neural network by the Feedforward and Backpropagation processes.

Obtain the affine factor matrices $\hat{\boldsymbol{U}}^t$ or $\hat{\boldsymbol{V}}^t$.

Denormalise $\hat{\boldsymbol{U}}^t$ or $\hat{\boldsymbol{V}}^t$.

Phase 3: Cross-Domain and Cross-System Recommendations

For CDR, fix $\hat{\boldsymbol{U}}^t$ and train $\hat{\boldsymbol{V}}^t$ from $\boldsymbol{R}^t$ by using the MF model in Phase 1.

For CSR, fix $\hat{\boldsymbol{V}}^t$ and train $\hat{\boldsymbol{U}}^t$ from $\boldsymbol{R}^t$ by using the MF model in Phase 1.

Obtain the predicted ratings $\hat{\boldsymbol{R}}^t = \hat{\boldsymbol{U}}^t[\hat{\boldsymbol{V}}^t]^\top$ for the target domain or system.

return $\mathcal{V}_i$.

objective function is represented as follows:

$$\min_{U,V,\Theta} \Bigg(\sum_{u_i \in \mathcal{U}, v_j \in \mathcal{V}} \ell(U_i, V_j) + \lambda(\|U\|_F^2) + \|V\|_F^2) \Bigg), \tag{3.1}$$

where $\ell(U_i, V_j) = \sum_{h=1}^{p-1} max(0, 1 - (I_{ij}^h(\theta_{ij} - U_i^\top V_j)))$. $I_{ij}^h = +1$ for $R_h \geq R_{ij}$. Here, R_h represents the h-th rating, otherwise $I_{ij}^h = -1$. $\|\cdot\|_F$ represents the Frobenius norm and the user-specific threshold θ_{ij} is indicated in [104]. The parameters are trained by optimising the objective function via gradient descent.

3.3.1.2 Probabilistic Matrix Factorisation

This is a probabilistic model with Gaussian observation noise, and the core concept of PMF is maximising the conditional distribution over the observed ratings. The objective function of PMF model is:

$$\min_{U,V}\Bigg(\sum_{i=1}^{n}\sum_{j=1}^{m}\|I_{ij}\cdot(R_{ij}-U_i^{\top}V_j)\|_F^2 \\ +\lambda_U\|U\|_F^2+\lambda_V\|V\|_F^2\Bigg), \tag{3.2}$$

where $\lambda_U=\sigma^2/\sigma_U^2$, $\lambda_V=\sigma^2/\sigma_V^2$, and I_{ij} is the indicator variable. $I_{ij}=1$ represents that user u_i has rated on item v_j; otherwise $I_{ij}=0$. The parameters can be trained by minimising the objective function via gradient descent.

3.3.2 Ranking-Oriented Matrix Factorisation

3.3.2.1 Bayesian Personalised Ranking Model

The BPR is a generic optimisation benchmark that is used for personalised ranking and it creates a set of triples $D_s : U\times V\times V$ by:

$$D_s := \{(u_i,v_j,v_l)|R_{ij}>R_{il}\}. \tag{3.3}$$

To reduce the ranking error between predicted and observed ratings, BPR optimises the following objective function as so:

$$\min_{U,V}\Bigg(\sum_{(u_i,v_j,v_l)\in D_s}-ln\sigma(U_i^{\top}V_j-U_i^{\top}V_l) \\ +\lambda_U\|U\|_F^2+\lambda_V\|V\|_F^2\Bigg), \tag{3.4}$$

the parameters are also trained by gradient descent.

3.4 Phase 2: The DNN Mapping

The user and item latent factor matrices $\{\boldsymbol{U}^s,\boldsymbol{U}^t,\boldsymbol{V}^s,\boldsymbol{V}^t\}$ can be learned by the abovementioned MF models. Next, a fully connected DNN is developed to represent the relationship of latent factors between two domains or systems (i.e., DNN mapping).

As mentioned in Section 2, both enforcing $\{\boldsymbol{U}^t, \boldsymbol{V}^t\}$ to be the same as $\{\boldsymbol{U}^s, \boldsymbol{V}^s\}$ [139] and mapping $\{\boldsymbol{U}^s, \boldsymbol{V}^s\}$ to fit $\{\boldsymbol{U}^t, \boldsymbol{V}^t\}$ [71] are neither effective nor reasonable because the accuracies of user and item latent factors mainly depend on their sparsity degrees. More importantly, a common entity in the source domain or system may be sparser than the entity in the target domain or system. This means that the latent factors of the entity in the source domain or system are less accurate than those of the entity in the target domain or system. Therefore, more reasonable benchmark factor matrices $\boldsymbol{U}^b$ and $\boldsymbol{V}^b$ are generated by integrating latent factors and by considering the sparsity degrees of individual users and items in both the source and target domains or systems.

3.4.1 The Generation of Benchmark Factors

First, the *common users* $\mathcal{CU}$ are extracted from two different domains for CDR and the *common items* $\mathcal{CV}$ are extracted from two different systems for CSR.

Then, the sparsity degrees of common entities (either users or items) are defined in the different domains or systems below.

Definition 2. Sparsity Degrees of Common Entities.

- For any common entity $e_i \in \mathcal{CU} \cup \mathcal{CV}$,
- Given the total numbers of ratings of e_i in the source and target domains or systems, n_i^s and n_i^t,
- The sparsity degrees α_i^s and α_i^t of the entity e_i in the source and target domains or systems are calculated as

$$\alpha_i^s = \frac{n_i^t}{(n_i^s + n_i^t)}, \quad \alpha_i^t = 1 - \alpha_i^s. \tag{3.5}$$

To generate more reasonable benchmark factors for the following DNN mapping process, both $\{\boldsymbol{U}^s, \boldsymbol{V}^s\}$ and $\{\boldsymbol{U}^t, \boldsymbol{V}^t\}$ are non-negligible factors. Therefore, based on the concept of feature combination that was introduced in [11], for each common user $u_i \in \mathcal{CU}$, the benchmark factor vector $\boldsymbol{CU}_i^b$ can be calculated as follows:

$$\boldsymbol{CU}_i^b = (1 - \alpha_i^s) \cdot \boldsymbol{U}_i^s + (1 - \alpha_i^t) \cdot \boldsymbol{U}_i^t. \tag{3.6}$$

According to Equation (3.6), the smaller that the sparsity degrees α_i^s and α_i^t are, then the more accurate their corresponding latent factor vectors $\boldsymbol{U}_i^s$ and $\boldsymbol{U}_i^t$ will be — and thus, the more these vectors are considered in generating a benchmark factor vector $\boldsymbol{CU}_i^b$.

Likewise, for CSR, the benchmark factor matrix $\boldsymbol{CV}^b$ can be obtained for the common items.

Next, the *different users* $\mathcal{DU}^t = \mathcal{U}^t - \mathcal{CU}$ are identified in the target domain for CDR, and the *different items* $\mathcal{DV}^t = \mathcal{V}^t - \mathcal{CV}$ are identified in the target system for CSR. Later on, the cosine similarity is employed to measure the similarities between common and different entities. For each user $u_i \in \mathcal{DU}^t$, the top-k similar users $\mathcal{SU}_i$ from $\mathcal{CU}$ are chosen. Similarly, for each item $v_i \in \mathcal{DV}^t$, the top-k similar items $\mathcal{SV}_i$ from $\mathcal{CV}$ are chosen. Based on these top-k similar entities, the sparsity degrees of different entities are defined as follows:

Definition 3. Sparsity degrees of different entities

- For any different entity $e_i \in \mathcal{DU}^t \cup \mathcal{DV}^t$,

- Given the total number of ratings of e_i in the target domain or system, n_i^s, and the average number of ratings of e_i's top-k similar entities in the source domain or system, sn_i^s,

- The sparsity degree β_i^t of entity e_i in the target domain or system is calculated as

$$\beta_i^t = \frac{sn_i^s}{(n_i^t + sn_i^s)}. \tag{3.7}$$

Therefore, for each different user $u_i \in \mathcal{DU}^t$, the benchmark factor vector $\boldsymbol{DU}_i^b$ can be calculated as:

$$\begin{aligned} \boldsymbol{DU}_i^b &= (1-\beta_i^t)\cdot \boldsymbol{U}_i^t + \beta_i^t \cdot \boldsymbol{SU}_i, \text{where} \\ \boldsymbol{SU}_i &= \frac{\sum\limits_{u_j \in \mathcal{SU}_i} sim(u_i,u_j)\cdot \boldsymbol{U}_j^s}{\sum\limits_{u_j \in \mathcal{SU}_i} sim(u_i,u_j)}. \end{aligned} \tag{3.8}$$

Similarly, for CSR, the benchmark factor matrix $\boldsymbol{DV}^b$ can be obtained for different items.

Finally, we have $\boldsymbol{U}^b = \boldsymbol{CU}^b \cup \boldsymbol{DU}^b$ and $\boldsymbol{V}^b = \boldsymbol{CV}^b \cup \boldsymbol{DV}^b$.

3.4.2 The Mapping Process

3.4.3 Normalisation

The abovementioned feature combination method can be used to obtain the benchmark factor matrices $\{\boldsymbol{U}^b, \boldsymbol{V}^b\}$. The latent factor matrices $\{\boldsymbol{U}^t, \boldsymbol{V}^t\}$ and the benchmark factor matrices $\{\boldsymbol{U}^b, \boldsymbol{V}^b\}$ are normalised into the range $[-1, 1]$ by using the mapminmax function.

3.4.4 Mapping Process

As shown in Phase 2 of Figure 3.1, a fully connected DNN is employed to map $\boldsymbol{U}^t$ to fit $\boldsymbol{U}^b$ for CDR and $\boldsymbol{V}^t$ to fit $\boldsymbol{V}^b$ for CSR, respectively. Since the mapping processes for CDR and CSR are similar, CDR is selected as the example for introducing the DNN mapping process. In general, to minimise the mapping loss for CDR, the process of training mapping parameters can be changed into the following minimisation problem:

$$\min_{\Theta} \ell(h(\boldsymbol{U}^t;\Theta), \boldsymbol{U}^b), \tag{3.9}$$

where $h(\cdot)$ is a DNN mapping function that is introduced below, and the loss function $\ell(\cdot)$ is the square loss.

This detailed training process is divided into the two steps of Feedforward and Backpropagation, as outlined below:

- **Feedforward:** Each latent factor vector is denoted as a low-dimensional vector, and each input vector is mapped into a hidden vector in each layer. Let x_j denote the input vector, W_j denote the weight vector, b_j denote the bias term and y_j denote the output vector for the j-th hidden layer, $j = 1, ..., d$. Therefore, for $\boldsymbol{U}_i^t \subset \boldsymbol{U}^t$, we have

$$\begin{aligned} & x_1 = \boldsymbol{U}_i^t, \\ & y_j = f(W_j \cdot x_j + b_j), \qquad j = 1, ..., d-1, \\ & h(\boldsymbol{U}_i^t; \Theta) = f(W_d \cdot x_d + b_d), \qquad \Theta = \{W; b\}, \end{aligned} \tag{3.10}$$

  and the tan-sigmoid function is chosen as the activation function — that is, $f(x) = \frac{2}{(1+\exp -2x)} - 1$.

- **Backpropagation:** According to the chain rule, the parameters are recursively updated by computing the gradients of all inputs, parameters and intermediates, as introduced in [93].

The mapping process is similar for CSR, involving the replacement of $\{\boldsymbol{U}^t, \boldsymbol{U}^b\}$ with $\{\boldsymbol{V}^t, \boldsymbol{V}^b\}$ as the input for the DNN.

3.4.5 Denormalisation

After the DNN mapping, the affine factor matrix $\hat{\boldsymbol{U}}^t$ for CDR and $\hat{\boldsymbol{V}}^t$ for CSR is obtained. Finally, the affine factor matrix is denormalised into the range of the original latent factor matrix by reversing the mapminmax function.

3.5 Phase 3: Cross-Domain and Cross-System Recommendations

3.5.1 Cross-Domain Recommendation

In most cases, the items in the source and target domains are definitely different. Therefore, with the DNN mapping for CDR, the affine factor matrix $\hat{\boldsymbol{U}}^t$ can be obtained. However, the original $\boldsymbol{V}^t$ has yet to be improved. To this end, we fix $\hat{\boldsymbol{U}}^t$ and only update $\boldsymbol{V}^t$ by using the MF models to generate $\hat{\boldsymbol{V}}^t$.

3.5.2 Cross-System Recommendation

Similarly, either the users in the source and target systems are totally different, or it is difficult to determine whether they are the same. Therefore, for CSR, we first obtain $\hat{\boldsymbol{V}}^t$ using the DNN mapping, then fix $\hat{\boldsymbol{V}}^t$ and obtain $\hat{\boldsymbol{U}}^t$ by using the MF models.

Finally, based on $\hat{\boldsymbol{U}}^t$ and $\hat{\boldsymbol{V}}^t$, matched items $\mathcal{V}_i \subseteq \mathcal{V}^t$ can be recommended to target users $u_i \in \mathcal{U}^t$ for both CDR and CSR.

3.6 Experiments on DCDCSR

Extensive experiments are conducted on three real-world datasets, which aim to answer the following questions:

- **Q1:** How does the dimension K of the latent factors affect the efficiency of the DCDCSR framework? (in Result 1)

- **Q2:** How does the DCDCSR approach outperform the state-of-the-art approaches for both cross-domain and cross-system recommendations? (in Results 2 & 3)

Table 3.1: Experimental datasets for DCDCSR

Tasks	Cross-Domain			Cross-System		
Datasets	Douban			Netflix	MovieLens	Douban*
Domains	Movie	Book	Music	Movie	Movie	Movie
#Users	3,982	3,032	1,983	59,688	138,493	500
#Items	90,553	87,848	88,986	17,434	27,278	90,553
#Ratings	2,326,913	239,330	242,013	2,000,000	20,000,263	48,619

3.6.1 Experimental Settings

3.6.1.1 Datasets

Three real-world datasets are used in the experiments — namely, the two public benchmark datasets of Netflix Prize[1] and MovieLens 20M[2] and a Douban dataset that was crawled from the Douban website. Since MovieLens 20M contains more than 20 million ratings (it is relatively richer), for the diversity of the experiments, a subset was extracted from the Netflix Prize, which has a smaller scale of ratings (2 million). The details of these three datasets are shown in Table 4.1.

For the CDR experiments, DoubanMovie was taken as the source domain corresponding to the target domains DoubanBook and DoubanMusic. For the CSR experiments, Netflix and MovieLens were taken as the source systems, and a subset Douban*Movie from DoubanMovie was extracted as the target system. The numbers of common items for Netflix-Douban* and for MovieLens-Douban* are 3,700 and 5,712, respectively. For the Douban dataset, the numbers of Movie-Music and Movie-Book common users are 295 and 379, respectively.

In the experiments, each dataset was split into a training set (80%) with the early ratings and a test set (20%) with the later ratings. The sequences of ratings and latent factors may have slightly affected the performances of MF and mapping, respectively. Therefore, the average results of five random times were reported.

[1] https://www.kaggle.com/netflix-inc/netflix-prize-data
[2] https://www.kaggle.com/grouplens/movielens-20m-dataset

3.6.1.2 Parameter Setting

The dimension K of the latent factor was set as 10, 20, 50 and 100, respectively. To generate the benchmark factors, $k = 5$ was set for top-k similar items or users. For the DNN, the depth of the hidden layers d was set to five because when $d > 5$, the performances of the methods almost do not change. The dimension of the input and output of the DNN was set to K, and the number of hidden nodes was set to $1.5 \times K$. The parameters were randomly initialised, as suggested in [30] — that is, $W \sim U[-\frac{1}{\sqrt{2K}}, \frac{1}{\sqrt{2K}}]$. Additionally, the batch size was set to 32, and the learning rate was set to 0.005.

3.6.1.3 Experimental Tasks and Evaluation Metrics

In total, two CDR tasks and two CSR tasks were designed as follows:

- **Task 1:** DoubanMovie $\rightarrow$ DoubanBook (for CDR),

- **Task 2:** DoubanMovie $\rightarrow$ DoubanMusic (for CDR),

- **Task 3:** Netflix $\rightarrow$ Douban*Movie (for CSR),

- **Task 4:** MovieLens $\rightarrow$ Douban*Movie (for CSR).

The mean absolute error (MAE) and the root mean squared error (RMSE) were used as metrics to evaluate the recommendation performance. These metrics are commonly used in the literature for both CDR and CSR [84, 139].

3.6.1.4 Comparison Methods

In the experiments, the DCDCSR framework was implemented into three methods by applying MMMF, PMF and BPR as the MF models — that is, MMMF _DCDCSR, PMF_DCDCSR and BPR_DCDCSR.

The study's three DCDCSR methods were compared with the following seven methods that were implemented from three representative models:

- *Bayesian personalised ranking model* (BPR) [91]: BPR is a ranking-oriented MF model. It was chosen as a conventional baseline method running on the target domain and system, which does not take any cross-domain or cross-system strategies.

- *Active transfer learning framework* (ATL) [139]: This is a state-of-the-art framework that uses transfer learning (TL). It offers three methods. In this thesis's experiments, the two well-performing methods of MMMF_TL and PMF_TL were chosen.

- *Embedding and mapping framework* (EMCDR) [71]: This is a state-of-the-art framework that uses linear matrix translation (LIN) and MLP. It adopts PMF and BPR as its MF models and maps the latent factors across domains or systems with both LIN and MLP (2×2). Therefore, this framework offers four methods — namely, MF_EMCDR_LIN, MF_EMCDR_MLP, BPR_EMCDR_LIN and BPR_EMCDR_MLP, which are all compared in the experiments.

3.6.2 Performance Comparison and Analysis

All the experimental results are presented in Tables 3.2 - 3.5.

3.6.2.1 Result 1: The Effect of Latent Factor Dimension

To answer question **Q1**, this thesis investigates how the performance of the DCDCSR framework is affected by the dimension K of the latent factors. From Tables 3.2-3.5, it can be observed that when $K = 10$ or 20, the performances of DCDCSR methods generally increase (i.e., the MAE and RMSE decrease) with K. However, when $K = 50$, no significant improvement is observed in the performance. Moreover, when $K =$

Table 3.2: The experimental results of CDR (Part 1)

		Cross-Domain Recommendation (CDR)			
		Task 1		**Task 2**	
		MAE	RMSE	MAE	RMSE
K=10	BPR	0.7187 (± 0.0011)	0.9386 (± 0.0014)	0.7231 (± 0.0012)	0.9416 (± 0.0017)
	MMMF_TL	0.7001 (± 0.0009)	0.9128 (± 0.0007)	0.6978 (± 0.0006)	0.9093 (± 0.0005)
	PMF_TL	0.7022 (± 0.0016)	0.9187 (± 0.0006)	0.7077 (± 0.0008)	0.9097 (± 0.0005)
	MF_EMCDR_LIN	0.7065 (± 0.0003)	0.9103 (± 0.0006)	0.7024 (± 0.0012)	0.9163 (± 0.0004)
	MF_EMCDR_MLP	0.7011 (± 0.0015)	0.9071 (± 0.0009)	0.7022 (± 0.0008)	0.9045 (± 0.0012)
	BPR_EMCDR_LIN	0.7084 (± 0.0012)	0.9111 (± 0.0006)	0.7065 (± 0.0005)	0.9105 (± 0.0013)
	BPR_EMCDR_MLP	0.7061 (± 0.0005)	0.9054 (± 0.0005)	0.6987 (± 0.0003)	0.9055 (± 0.0008)
	MMMF_DCDCSR	0.7041 (± 0.0005)	0.8971 (± 0.0004)	0.6992 (± 0.0003)	0.8875 (± 0.0002)
	PMF_DCDCSR	0.7037 (± 0.0005)	0.8965 (± 0.0003)	0.6996 (± 0.0004)	**0.8866 (± 0.0002)**
	BPR_DCDCSR	**0.6943 (± 0.0003)**	**0.8881 (± 0.0006)**	**0.6971 (± 0.0008)**	0.8872 (± 0.0004)
K=20	BPR	0.7146 (± 0.0014)	0.9292 (± 0.0007)	0.7234 (± 0.0011)	0.9352 (± 0.0006)
	MMMF_TL	0.7068 (± 0.0004)	0.9146 (± 0.0008)	0.7109 (± 0.0003)	0.9104 (± 0.0002)
	PMF_TL	0.7017 (± 0.0003)	0.9188 (± 0.0008)	0.7176 (± 0.0004)	0.9244 (± 0.0006)
	MF_EMCDR_LIN	0.7015 (± 0.0008)	0.9070 (± 0.0006)	0.7021 (± 0.0006)	0.9076 (± 0.0019)
	MF_EMCDR_MLP	0.7021 (± 0.0003)	0.9095 (± 0.0005)	0.7001 (± 0.0003)	0.9095 (± 0.0005)
	BPR_EMCDR_LIN	0.7041 (± 0.0009)	0.9174 (± 0.0005)	0.7021 (± 0.0008)	0.9147 (± 0.0012)
	BPR_EMCDR_MLP	0.7023 (± 0.0006)	0.9074 (± 0.0006)	0.7021 (± 0.0008)	0.9047 (± 0.0012)
	MMMF_DCDCSR	0.7001 (± 0.0002)	0.8876 (± 0.0004)	0.6987 (± 0.0003)	0.8866 (± 0.0003)
	PMF_DCDCSR	0.7003 (± 0.0004)	0.8872 (± 0.0005)	0.6985 (± 0.0003)	0.8879 (± 0.0004)
	BPR_DCDCSR	**0.6941 (± 0.0002)**	**0.8845 (± 0.0001)**	**0.6949 (± 0.0004)**	**0.8867 (± 0.0003)**

100, the performances depict a slight decline. This is because the number of DNN parameters geometrically increases by K. When $K = 100$, while the training data remains the same, the performance of the DNN mapping declines slightly.

3.6.2.2 Result 2: Cross-Domain Recommendation (Tasks 1 & 2)

To answer question **Q2**, this thesis compares the performances of its methods and the seven comparison methods in the CDR tasks (Tasks 1 & 2). From Tables 3.2-3.5, it can be observed that, for the CDR tasks, MMMF_DCDCSR does not perform as well as PMF_DCDCSR and BPR_DCDCSR. This is because its MF model cannot effectively learn a predicted matrix $\hat{R}$ by maximising the predictive trace margin in the target domains DoubanBook and DoubanMusic.

Specifically, in terms of MAE, PMF_DCDCSR performed the best. It also outper-

Table 3.3: The experimental results of CDR (Part 2)

		Cross-Domain Recommendation (CDR)			
		Task 1		Task 2	
		MAE	RMSE	MAE	RMSE
K=50	BPR	0.7115 (± 0.0014)	0.9413 (± 0.0011)	0.7252 (± 0.0005)	0.9464 (± 0.0008)
	MMMF_TL	0.7062 (± 0.0010)	0.9189 (± 0.0009)	0.7143 (± 0.0007)	0.9132 (± 0.0004)
	PMF_TL	0.7022 (± 0.0005)	0.9203 (± 0.0004)	0.7121 (± 0.0012)	0.9287 (± 0.0007)
	MF_EMCDR_LIN	0.7051 (± 0.0003)	0.9080 (± 0.0002)	0.7021 (± 0.0008)	0.9082 (± 0.0006)
	MF_EMCDR_MLP	0.7065 (± 0.0005)	0.9114 (± 0.0006)	0.7076 (± 0.0004)	0.9086 (± 0.0008)
	BPR_EMCDR_LIN	0.7055 (± 0.0007)	0.9086 (± 0.0004)	0.7020 (± 0.0002)	0.9084 (± 0.0003)
	BPR_EMCDR_MLP	0.6917 (± 0.0004)	0.8994 (± 0.0005)	0.6987 (± 0.0004)	0.9003 (± 0.0001)
	MMMF_DCDCSR	0.7003 (± 0.0002)	0.8880 (± 0.0003)	0.6988 (± 0.0001)	0.8889 (± 0.0001)
	PMF_DCDCSR	**0.6941 (± 0.0001)**	0.8871 (± 0.0002)	**0.6918 (± 0.0004)**	0.8925 (± 0.0002)
	BPR_DCDCSR	0.6954 (± 0.0002)	**0.8862 (± 0.0003)**	0.6957 (± 0.0002)	**0.8874 (± 0.0002)**
K=100	BPR	0.7199 (± 0.0005)	0.9332 (± 0.0011)	0.7303 (± 0.0005)	0.9396 (± 0.0005)
	MMMF_TL	0.7104 (± 0.0003)	0.9191 (± 0.0002)	0.7124 (± 0.0003)	0.9241 (± 0.0001)
	PMF_TL	0.7089 (± 0.0005)	0.9213 (± 0.0004)	0.7071 (± 0.0008)	0.9207 (± 0.0005)
	MF_EMCDR_LIN	0.6994 (± 0.0012)	0.9094 (± 0.0009)	0.7026 (± 0.0009)	0.9097 (± 0.0003)
	MF_EMCDR_MLP	0.7014 (± 0.0004)	0.9001 (± 0.0004)	0.7011 (± 0.0004)	0.8991 (± 0.0005)
	BPR_EMCDR_LIN	0.6985 (± 0.0004)	0.9098 (± 0.0001)	0.7030 (± 0.0008)	0.9099 (± 0.0003)
	BPR_EMCDR_MLP	0.7024 (± 0.0006)	0.8981 (± 0.0003)	0.7089 (± 0.0001)	0.8972 (± 0.0002)
	MMMF_DCDCSR	0.7004 (± 0.0003)	0.8904 (± 0.0002)	0.7005 (± 0.0002)	0.8932 (± 0.0003)
	PMF_DCDCSR	0.6986 (± 0.0001)	0.8895 (± 0.0004)	**0.6942 (± 0.0001)**	0.8931 (± 0.0001)
	BPR_DCDCSR	**0.6971 (± 0.0001)**	**0.8882 (± 0.0002)**	0.6998 (± 0.0003)	**0.8904 (± 0.0001)**

formed the seven comparison methods by an average of 1.42%, ranging from 0.94% to 3.57%. Moreover, in terms of RMSE, BPR_DCDCSR performed the best. It also outperformed the seven comparison methods by an average of 2.6%, ranging from 1.66% to 5.41%. Compared to all seven comparison methods, this thesis's methods clearly performed better because its sparsity-guided DNN mapping process can map the latent factors across domains more accurately.

3.6.2.3 Result 3: Cross-System Recommendation (Tasks 3 & 4)

To answer question **Q2**, this experiment compares the performances of its methods and the seven comparison methods in the CSR tasks (Tasks 3 & 4). From Tables 3.2-3.5, it can be observed that except when $K = 10$ in Task 4, BPR_DCDCSR outperformed MMMF_DCDCSR and PMF_DCDCSR. This is because its MF model can

Table 3.4: The experimental results of CSR (Part 1)

		Cross-System Recommendation (CSR)			
		Task 3		**Task 4**	
		MAE	RMSE	MAE	RMSE
K=10	BPR	0.7524 (± 0.0014)	0.9628 (± 0.0016)	0.7524 (± 0.0014)	0.9628 (± 0.0016)
	MMMF_TL	0.7162 (± 0.0012)	0.8951 (± 0.0003)	0.7090 (± 0.0007)	0.8997 (± 0.0003)
	PMF_TL	0.7031 (± 0.0008)	0.8913 (± 0.0012)	0.7120 (± 0.0003)	0.9030 (± 0.0007)
	MF_EMCDR_LIN	0.7096 (± 0.0008)	0.9113 (± 0.0007)	0.7340 (± 0.0009)	0.9326 (± 0.0007)
	MF_EMCDR_MLP	0.7087 (± 0.0008)	0.9049 (± 0.0005)	0.7045 (± 0.0004)	0.9062 (± 0.0005)
	BPR_EMCDR_LIN	0.7038 (± 0.0004)	0.9035 (± 0.0003)	0.7080 (± 0.0005)	0.9043 (± 0.0006)
	BPR_EMCDR_MLP	0.6995 (± 0.0005)	0.8994 (± 0.0003)	0.6991 (± 0.0002)	0.8994 (± 0.0005)
	MMMF_DCDCSR	0.6998 (± 0.0003)	0.8865 (± 0.0002)	0.6994 (± 0.0005)	0.8836 (± 0.0004)
	PMF_DCDCSR	0.6838 (± 0.0012)	0.8681 (± 0.0011)	**0.6753 (± 0.0006)**	**0.8659 (± 0.0007)**
	BPR_DCDCSR	**0.6786 (± 0.0007)**	**0.8651 (± 0.0008)**	0.6854 (± 0.0014)	0.8712 (± 0.0009)
K=20	BPR	0.7432 (± 0.0012)	0.9532 (± 0.0014)	0.7432 (± 0.0012)	0.9532 (± 0.0014)
	MMMF_TL	0.6915 (± 0.0002)	0.8922 (± 0.0003)	0.7026 (± 0.0003)	0.8986 (± 0.0002)
	PMF_TL	0.7024 (± 0.0003)	0.8969 (± 0.0002)	0.7057 (± 0.0003)	0.9012 (± 0.0003)
	MF_EMCDR_LIN	0.7027 (± 0.0005)	0.9074 (± 0.0013)	0.6977 (± 0.0015)	0.9032 (± 0.0002)
	MF_EMCDR_MLP	0.6995 (± 0.0003)	0.8995 (± 0.0003)	0.6993 (± 0.0005)	0.8995 (± 0.0005)
	BPR_EMCDR_LIN	0.7060 (± 0.0007)	0.9024 (± 0.0005)	0.6949 (± 0.0006)	0.9012 (± 0.0008)
	BPR_EMCDR_MLP	0.6991 (± 0.0005)	0.8993 (± 0.0003)	0.6995 (± 0.0002)	0.8999 (± 0.0002)
	MMMF_DCDCSR	0.7004 (± 0.0003)	0.8875 (± 0.0004)	0.7012 (± 0.0001)	0.8816 (± 0.0004)
	PMF_DCDCSR	0.6880 (± 0.0001)	0.8609 (± 0.0006)	0.6805 (± 0.0004)	0.8654 (± 0.0001)
	BPR_DCDCSR	**0.6723 (± 0.0002)**	**0.8556 (± 0.0008)**	**0.6780 (± 0.0003)**	**0.8601 (± 0.0002)**

create numerous triples in the target system Douban* to train the parameters, which can generate relatively accurate latent factors.

Specifically, in terms of MAE, BPR_DCDCSR outperformed all seven comparison methods by an average of 4.20%, ranging from 3.00% to 9.00%. Moreover, in terms of RMSE, BPR_DCDCSR outperformed all seven comparison methods by an average of 4.46%, ranging from 3.43% to 9.08%. Compared to all seven comparison methods, this study's methods clearly performed better because its sparsity-guided DNN mapping process maps the latent factors across systems more accurately.

Further, the proposed DCDCSR methods delivered more improvements in terms of MAE and RMSE in Result 3 when compared to Result 2. This is because these methods can effectively use the ratings of the source systems MovieLens and Netflix when the ratings are much richer than those of the target system Douban*.

Table 3.5: The experimental results of CSR (Part 2)

		Cross-System Recommendation (CSR)			
		Task 3		**Task 4**	
		MAE	RMSE	MAE	RMSE
K=50	BPR	0.7252 (± 0.0012)	0.9364 (± 0.0018)	0.7252 (± 0.0012)	0.9364 (± 0.0018)
	MMMF_TL	0.6899 (± 0.0002)	0.8851 (± 0.0003)	0.6948 (± 0.0003)	0.8975 (± 0.0004)
	PMF_TL	0.7011 (± 0.0012)	0.8954 (± 0.0010)	0.7021 (± 0.0007)	0.8974 (± 0.0012)
	MF_EMCDR_LIN	0.7095 (± 0.0014)	0.9062 (± 0.0005)	0.7012 (± 0.0007)	0.9055 (± 0.0009)
	MF_EMCDR_MLP	0.6997 (± 0.0005)	0.8997 (± 0.0004)	0.6993 (± 0.0006)	0.8995 (± 0.0002)
	BPR_EMCDR_LIN	0.7013 (± 0.0004)	0.9034 (± 0.0012)	0.6983 (± 0.0005)	0.9016 (± 0.0008)
	BPR_EMCDR_MLP	0.7022 (± 0.0011)	0.8981 (± 0.0005)	0.6995 (± 0.0004)	0.9003 (± 0.0001)
	MMMF_DCDCSR	0.6924 (± 0.0007)	0.8856 (± 0.0005)	0.6935 (± 0.0002)	0.8746 (± 0.0003)
	PMF_DCDCSR	0.8655 (± 0.0009)	0.6794 (± 0.0014)	0.8636 (± 0.0011)	
	BPR_DCDCSR	**0.6712 (± 0.0008)**	**0.8555 (± 0.0007)**	**0.6595 (± 0.0003)**	**0.8564 (± 0.0002)**
K=100	BPR	0.7334 (± 0.0012)	0.9321 (± 0.0004)	0.7334 (± 0.0012)	0.9321 (± 0.0004)
	MMMF_TL	0.6931 (± 0.0002)	0.8772 (± 0.0003)	0.6948 (± 0.0003)	0.8997 (± 0.0002)
	PMF_TL	0.7020 (± 0.0003)	0.8966 (± 0.0002)	0.6995 (± 0.0003)	0.8954 (± 0.0005)
	MF_EMCDR_LIN	0.7045 (± 0.0005)	0.9060 (± 0.0003)	0.6961 (± 0.0002)	0.9082 (± 0.0012)
	MF_EMCDR_MLP	0.7001 (± 0.0002)	0.9008 (± 0.0008)	0.6998 (± 0.0012)	0.9004 (± 0.0001)
	BPR_EMCDR_LIN	0.7077 (± 0.0011)	0.9072 (± 0.0002)	0.7017 (± 0.0006)	0.9099 (± 0.0008)
	BPR_EMCDR_MLP	0.6999 (± 0.0005)	0.9000 (± 0.0008)	0.6995 (± 0.0002)	0.9003 (± 0.0006)
	MMMF_DCDCSR	0.6915 (± 0.0004)	0.8798 (± 0.0002)	0.6865 (± 0.0009)	0.8769 (± 0.0004)
	PMF_DCDCSR	0.6852 (± 0.0012)	0.8718 (± 0.0009)	0.6814 (± 0.0003)	0.8665 (± 0.0004)
	BPR_DCDCSR	**0.6745 (± 0.0008)**	**0.8612 (± 0.0011)**	**0.6678 (± 0.0004)**	**0.8594 (± 0.0002)**

3.6.2.4 Experimental Summary

According to Result 1, question **Q1** can be answered as follows: In general, the performances of DCDCSR methods increase with the dimension K of the latent factors when $K \in \{10, 20\}$. However, when $K \in \{50, 100\}$, the performances display no significant improvement, and even decline slightly. According to Results 2 and 3, question **Q2** can be answered as follows: In general, the proposed DCDCSR methods outperformed all comparison methods for both CDR and CSR because its sparsity-guided DNN mapping process can map latent factors across domains or systems more accurately. Additionally, comparing Results 2 and 3 demonstrated that these methods can effectively utilise more rating data.

3.7 Summary

In this chapter, a Deep framework for both CDR and CSR, called DCDCSR, has been proposed, which is based on MF models and a fully connected Deep Neural Network (DNN). The DNN is applied to more accurately map the latent factors across domains or systems. In addition, the sparsity degrees of individual users and items are utilised in the source and target domains or systems to guide the DNN training process, which can effectively utilise more rating data. The superior performance of our model has been demonstrated by extensive experiments conducted on three real-world datasets.

Chapter 4

DTCDR: A Framework for Dual-Target Cross-Domain Recommendation

All these existing CDR approaches only focus on how to leverage the source domain to help improve the recommendation accuracy in the target domain, but not vice versa. Namely, they are single-target CDR approaches. However, each domain may be relatively richer in certain types of information (e.g., ratings, reviews, user profiles, item details and tags). Therefore, if such information can be leveraged effectively, it is possible to improve the recommendation performance in both domains simultaneously rather than in only a single-target domain.

Nevertheless, the novel dual-target CDR problem faces a new challenge, without any solution reported in the literature. This has introduced in Section 1.2.2 and expressed by **CH2**: *'how can a feasible framework for dual-target CDR be devised?'*. As an option, MTL has the potential for dual-target CDR because it aims to improve models' generalisation by leveraging the domain-specific information that is derived from the related recommendation tasks [94]. However, existing MTL-based recommendation approaches [3, 69] cannot be efficiently applied to dual-target CDR. This is because they heavily rely on the local feature representation and side information (additional information associated with the users and items) from a single domain, and such features and information in the sparser domain may be too sparse to support dual-target CDR. Additionally, MDR can also be considered another option. However, the proposed MDR models in [135, 80, 85, 137] achieved different goals — they either focused on improving the recommendation accuracies of specific or common users that were selected from multiple domains, or they only focused on improving the recommendation accuracy on a single-target domain. None of these possible methods can improve the recommendation accuracies of all users on multiple domains simultaneously. Therefore, existing MDR models cannot serve for dual-target CDR directly.

Moreover, to address the data sparsity problem, multi-source information — such as ratings, reviews, user profiles, item details and tags — that is derived from both domains should be leveraged to obtain more general user and item embeddings. Therefore, for dual-target CDR, there is another challenge, which has introduced in Section 1.2.3 and expressed by **CH3**: *'How can data richness and diversity be leveraged to generate more representative single-domain user/item embeddings for improving recommendation accuracy in both domains?'*.

To target the above challenges, we propose a novel framework for dual-target CDR in this chapter. To the best of our knowledge, this is the first work in the literature to propose the novel problem of dual-target CDR and provide a solution for it. The detailed framework is explained as follows.

4.1 Problem Statement

For readability purposes, the important notations of this chapter are listed in Tables A.1 and A.2. Based on these notations, the dual-target CDR is defined as follows:

Definition 4. Dual-target cross-domain recommendation

- Given two observed domains A and B which include the user ratings $\{R^a, R^b\}$, the user comments $\{C^a, C^b\}$, the user profiles $\{UP^a, UP^b\}$, and the item details $\{ID^a, ID^b\}$,

- The goal of dual-target CDR is to recommend the matched items $\mathcal{V}_i$ to any user

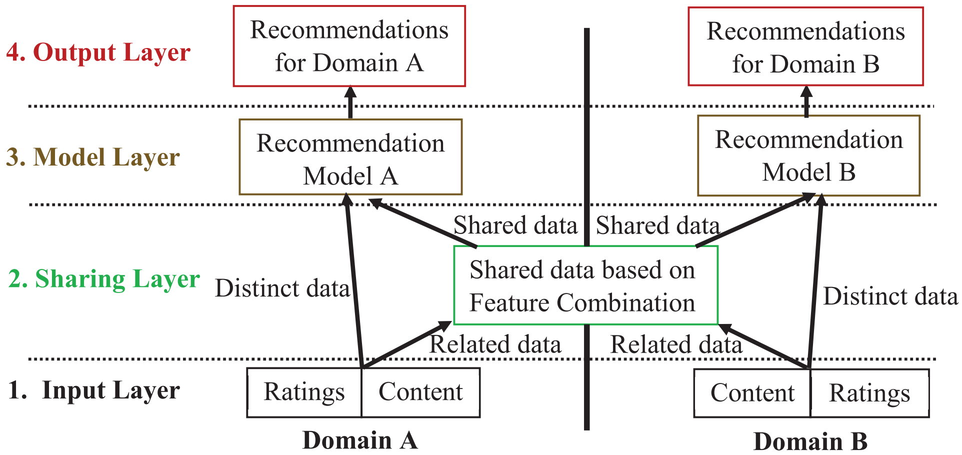


Figure 4.1: The general structure of the DTCDR framework

u_i on any of the two domains, rather than to only a user in the target domain for CDR.

In regard to the dual-target CDR problem, a certain degree of overlap is necessary between the users of different domains (i.e., common users), which can be used to link the two domains and share knowledge across them. This is common in the existing CDR approaches.

4.2 The General Framework for Dual-Target CDR

By targeting dual-target CDR, a general framework called DTCDR is proposed, as shown in Figure 4.1. The core idea of this framework is to utilise the data richness and diversity from the two domains A and B to improve the recommendation accuracies in both domains simultaneously. The framework contains four main parts: *Input Layer*, *Sharing Layer*, *Model Layer* and *Output Layer*.

These main parts can be summarised as follows:

- **Input Layer**: First, as mentioned in problem formulation,5 the input of the DTCDR framework contains users' explicit feedback (ratings and comments), user profiles and item details. Different domains may be richer in certain types of input data. According to different methods of embedding processing, the data of the *Input Layer* is divided into two types — the ratings and the content of users and items. The *Input Layer* contains the input data from domains A and B.

- **Sharing Layer**: On the top of the *Input Layer*, the *Sharing Layer* mainly focuses on combining the related data of common users from two domains by combining features and sharing them for the recommendation models of the two domains. The main goal of feature combination is to utilise the data richness and diversity of common users from both domains.

- **Model Layer**: In the *Model Layer*, both the distinct data from a domain and the shared data from the two domains are taken as the input, and the recommendation model is trained in each domain separately.

- **Output Layer**: Finally, the trained model can make recommendations for the corresponding domain in the *Output Layer*.

Note that this general framework can be extended for MDR if a certain degree of overlap could be found among multiple domains. This chapter only focuses on dual-target CDR with two domains.

Based on the core idea of the general DTCDR framework, this thesis proposes a specific MTL-based solution in the following sections.

4.3 Multi-Task Learning-Based Solution for the General DTCDR Framework

As shown in Figure 4.1, the recommendation models in different domains are parallel and closely related; the embeddings learned by ratings and content can thus be combined and shared by MTL. In this section, a specific MTL-based solution is posited for the DTCDR framework, as shown in Figure 4.2.

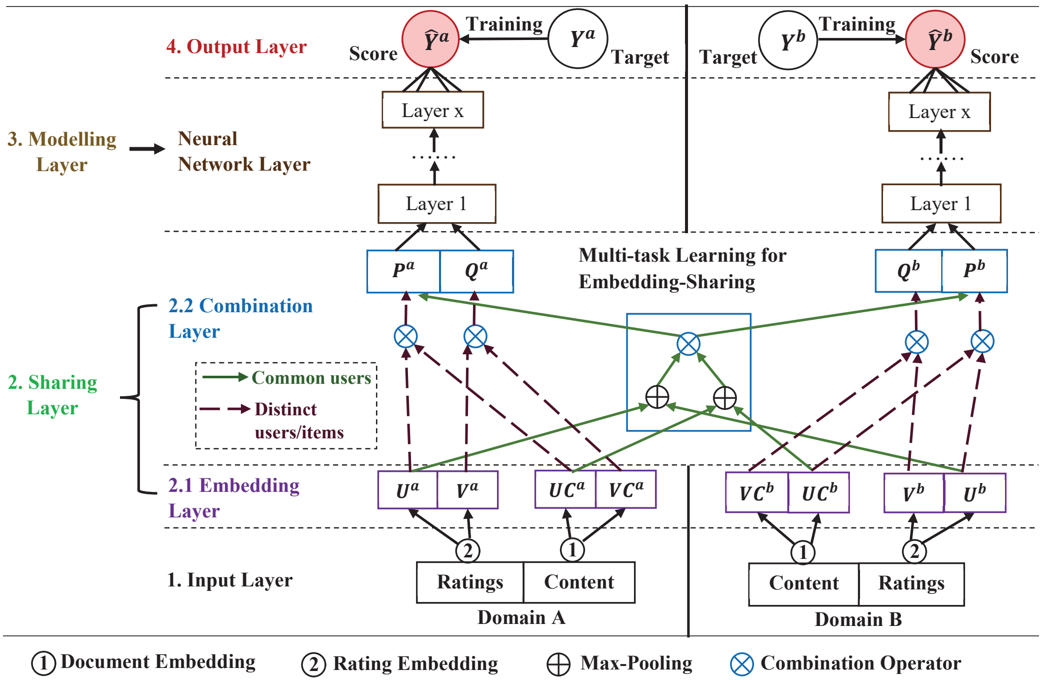


Figure 4.2: MTL-based solution for the DTCDR framework

Based on the general framework in Figure 4.1, the *Sharing Layer* is further divided into an *Embedding Layer* and a *Combination Layer*, and the *Model Layer* is implemented with the *Neural Network Layer*. Therefore, the specific MTL-based solution (see Figure 4.2) includes five layers: the *Input Layer*, *Embedding Layer*, *Combination Layer*, *Neural Network Layer* and *Output Layer*.

These layers can be described as follows:

- **Input Layer**: First, the *Input Layer* contains the input data (ratings and content) from domains A and B.

- **Embedding Layer**: Above the *Input Layer* is the *Embedding Layer*, in which document embedding UC is generated for users and VC for items, and rating embedding U is generated for users and V for items. The detailed embedding processes will be explained in Sections 4.3.1 and 4.3.2.

- **Combination Layer**: Above the *Embedding Layer* is the *Combination Layer*, which combines the document embedding and the rating embedding as the optimised embeddings P^a, Q^a, P^b, and Q^b for users and items in the two domains.

Specifically, *Max-Pooling* was first chosen to combine the embeddings of common users that were learned from different domains because different domains are richer in certain types of input data, and because we expect to remain the dominating factors of them. Then, to be adaptable to different dual-target C-DR scenarios, an effective embedding-sharing strategy was designed in which three representative combination operators were chosen to respectively combine the rating embedding and document embedding: *Concatenation* (Concat), *Max-Pooling* (MP) and *Average-Pooling* (AP). Concat can preserve all embeddings that are learned from content and ratings, MP tends to remain their remarkable factors and AP preserves the mean values of content and rating embeddings. These combination operators can utilise document and rating embeddings in diverse ways and make this thesis's models adaptable to different scenarios.

- **Neural Network Layer**: This layer is used to model a non-linear interaction relationship between users and items, which can represent a complex user-item interaction relationship. A conventional MLP is adopted in this layer.

- **Output Layer**: Based on the *Neural Network Layer*, P, Q are mapped to the predicted user-item interaction matrix $\hat{Y}$ in the *Output Layer*. The training process in this layer involves minimising the error between the predicted user-item interaction matrix $\hat{Y}$ and the observed user-item interaction matrix Y.

The specific MTL-based solution is presented in Algorithm 2, with details explained in the following sections.

This thesis's proposed DTCDR framework can also apply to CSR [139, 142] — in which the two systems have the same domain but different users, and they can thus contain common items only, such as DoubanMovie and MovieLens (see Task 3 in Section 4.4). Accordingly, in Figure 4.2, common users must be replaced with common items for supporting dual-target CSR.

Algorithm 2 MTL-based solution for the DTCDR framework

Require: (**Input Layer**) Two observed domains A and B, including the user ratings $\{R^a, R^b\}$, the user comments $\{C^a, C^b\}$, the user profiles $\{UP^a, UP^b\}$, the item details $\{ID^a, ID^b\}$, the number of training iterations num_iter, and the model type Mt (NeuMF_DTCDR or DMF_DTCDR).

Ensure: Recommend items $\mathcal{V}_i \subseteq \mathcal{V}$ to a target user u_i in any of the two domains.

1: # **Embedding Layer:**
2: Learn UC^a, VC^a from C^a, UP^a, and ID^a, by the document embedding model;
3: Pre-train U^a, V^a from R^a by rating embedding models;
4: Learn UC^b, VC^b from C^b, UP^b, and ID^b, by the document embedding model;
5: Pre-train U^b, V^b from R^b by rating embedding models;
6: **while** $epoch$ from 1 to num_iter **do**
7:     # **Combination Layer:**
8:     Get the common users $\mathcal{U}^{ac} = \mathcal{U}^{bc} = \mathcal{U}^a \bigcap \mathcal{U}^b$;
9:     Get the distinct users $\mathcal{U}^{ad} = \mathcal{U}^a - \mathcal{U}^{ac}$ in domain A;
10:     Get the distinct users $\mathcal{U}^{bd} = \mathcal{U}^b - \mathcal{U}^{bc}$ in domain B;
11:     $U^c = U^{ac} \oplus U^{bc}$;
12:     $UC^c = UC^{ac} \oplus UC^{bc}$;
13:     $P^a = [U^c \otimes UC^c; U^{ad} \otimes UC^{ad}]$;
14:     $P^b = [U^c \otimes UC^c; U^{bd} \otimes UC^{bd}]$;
15:     $Q^a = V^a \otimes VC^a$;
16:     $Q^b = V^b \otimes VC^b$;
17:     # **Neural Network (NN) Layers & Output Layer:**
18:     Train the NN in domain A and B by Equation (4.10);
19:     **if** Mt is NeuMF_DTCDR **then**
20:         Predict user-item interactions $\hat{Y}^a$ in domain A by Equation (4.11);
21:     **else**
22:         Predict user-item interactions $\hat{Y}^a$ in domain A by Equation (4.12);
23:     **end if**
24:     Repeat Lines 19 to 23 to predict user-item interactions $\hat{Y}^b$ in domain B;
25: **end whilereturn** $\mathcal{V}_i$ according to $\hat{Y}^a$ and $\hat{Y}^b$.

4.3.1 Document Embedding for Embedding Layer

In regard to the content information of each domain, multi-source content information (e.g., reviews, user profiles, item details and tags) of users and items was considered to generate the text vectors by using document embedding. Document embedding is used to map documents or paragraphs to text vectors. The representative work for document embedding is Doc2Vec [56], which contains two sub-models — distributed memory and distributed bag of words (DBOW). DBOW was chosen as the training algorithm because in the proposed framework, the text vectors should not be affected by the word order in each document (considering that DBOW does not preserve any word order).

The detailed process worked as described in the following: *first*, for user u_i, u_i's user profile up_i was combined with u_i's comments (reviews and tags) C_{i*} to obtain a document d_i, while for item v_j, its item detail id_j is combined with the comments C_{*j} on v_j to obtain a document d_{m+j}. *Then*, the natural language tool StanfordCoreNLP [72] is applied for cleaning text data and word segmentation of the documents $D = \{d_1, d_2, ..., d_{m+n}\}$. *Finally*, the documents D were mapped into corresponding text vectors UC and VC for users and items respectively by using the Doc2Vec model.

4.3.2 Rating Embedding for Embedding Layer

In regard to the rating information of each domain, the latent factors U and V were generated for users and items based on two popular neural network-based models (i.e., NeuMF or DMF). In fact, in Section 4.3.3, these two rating embedding models will be optimised and deeply integrated within the proposed framework. The NeuMF and DMF models will be briefly introduced here, and further details can be found in [38, 123]. The general objective function can be represented as follows:

$$\min \sum_{r \in R^+ \cup R^-} \ell(y, \hat{y}) + \lambda \Omega(\Theta), \tag{4.1}$$

where $\ell(*)$ denotes a loss function, R^+ denotes the observed ratings, R^- means all zero elements in R, $\hat{r}$ is the predicted rating for r, $\Omega(\Theta)$ is the regulariser and λ is a hyper-parameter that controls the importance of the regulariser. Note that in this study's experiments, a certain number of negative instances were sampled, denoted by $R^-_{sampled}$, to replace R^-, which is widely used [91, 38, 123].

4.3.2.1 Neural Matrix Factorization

NeuMF [38] only considers implicit feedback that is deserved from explicit ratings, and its user-item interaction can be represented as:

$$y_{ij} = \begin{cases} 1, & \text{if } r_{ij} \text{ is known;} \\ 0, & \text{otherwise.} \end{cases} \tag{4.2}$$

NeuMF employs a generalised MF (GMF) model and an MLP to learn the user-item interaction function to predict ratings. First, the mapping function of GMF is expressed as:

$$\phi^{GMF} = P^G \odot Q^G, \tag{4.3}$$

where P^G and Q^G are user and item latent factors that were generated by the GMF model and $\odot$ denotes the element-wise product of vectors.

The mapping function of MLP is expressed as:

$$\phi^{MLP} = f(W_L(...f(W_2 \begin{bmatrix} P^M \\ Q^M \end{bmatrix} + b_2)...) + b_L), \tag{4.4}$$

where P^M and Q^M are user and item latent factors that were generated by the first layer of the MLP model, f(*) is the activation function *ReLU* — that is, $f(x) = max(0, x)$, $W_2, ..., W_L$ and $b_2, ..., b_L$ are the weights and biases of the MLP model.

Based on Equations (4.3, 4.4), the NeuMF model predicts the interaction of user u_i on item v_j as follows:

$$\begin{aligned} \hat{y}_{ij} &= f(h^\top \begin{bmatrix} \phi_{ij}^{GMF} \\ \phi_{ij}^{MLP} \end{bmatrix}), \\ h &\leftarrow \begin{bmatrix} \alpha h^{GMF} \\ (1-\alpha) h^{MLP} \end{bmatrix}, \end{aligned} \tag{4.5}$$

where h^{GMF} and h^{MLP} denote the edge weights of the output layers of the GMF and

MLP models, respectively, and α is a hyper-parameter.

NeuMF chooses the cross-entropy loss as its loss function, which can be formulated as follows:

$$\ell(y, \hat{y}) = y \log \hat{y} + (1 - y) \log(1 - \hat{y}). \tag{4.6}$$

4.3.2.2 Deep Matrix Factorization

Compared to NeuMF, the DMF model [123] considers both implicit and explicit feedback, and its user-item interaction can be represented as:

$$y_{ij} = \begin{cases} r_{ij}, & \text{if } r_{ij} \text{ is known;} \\ 0, & \text{otherwise.} \end{cases} \tag{4.7}$$

The core idea of the DMF model is to evaluate the cosine similarities between the user and item latent factors that were learned by their own corresponding ratings.

The DMF model first predicts user and item latent factors $P = \{p_1, p_2, ..., p_m\}$, $Q = \{q_1, q_2, ..., q_n\}$ and then evaluates the cosine similarities between them as the predicted ratings $\hat{R}$. The details are as follows:

$$\begin{aligned} \hat{y}_{ij} = cosine(p_i, q_j) &= \frac{p_i^\top q_j}{||p_i||\,||q_j||}, \\ p_i &= f(...f(W_{U_2} f(r_{i*} W_{U_1}))), \\ q_j &= f(...f(W_{V_2} f(r_{*j} W_{V_1}))), \end{aligned} \tag{4.8}$$

where $f(*)$ is *ReLU* function, r_{i*} represents user u_i's ratings across all items, r_{*j} represents item v_j's ratings across all users, $W_{U_1}, W_{U_2}...$ and $W_{V_1}, W_{V_2}...$ are the weights of multilayer networks in the different layers for P and Q, respectively.

DMF improves the loss function of NeuMF (Equation (4.6)) and proposes a so-called normalised cross-entropy loss, which is formulated as follows:

$$\ell(y, \hat{y}) = \frac{y}{max(R)} \log \hat{y} + (1 - \frac{y}{max(R)}) \log(1 - \hat{y}), \tag{4.9}$$

where $max(R)$ is the maximum rating in a dataset (e.g., five for a five-star system).

4.3.3 Model Training

The DTCDR models were trained by the following objective function in domain A:

$$\begin{aligned} &\min_{P^a, Q^a, \Theta^a} \sum_{y \in Y^{a+} \cup Y^{a-}} \ell(y, \hat{y}) + \lambda(\|P^a\|_F^2 + \|Q^a\|_F^2), \\ &[P^a, Q^a] = [[U^c \otimes UC^c; U^{ad} \otimes UC^{ad}], [V^a \otimes VC^a]], \end{aligned} \tag{4.10}$$

where Θ^a is the parameter set for domain A and $\otimes$ is the combination operator. The predicted user-item interaction $\hat{y} \in \hat{Y}$ will be defined according to the specific rating embedding model in the following sections (see Equations (4.11) and (4.12)). Additionally, U^c and UC^c represent the rating embedding and document embedding of common users from different domains, while U^{ad}, UC^{ad}, V^a and VC^a represent the embeddings of distinct users and all items from domain A. Likewise, the objective function in domain B can be obtained. The detailed combination process for P^a, Q^a, P^b and Q^b is shown between Lines 8 to 16 in Algorithm 2.

4.3.3.1 A DTCDR model through NeuMF (NeuMF_DTCDR)

NeuMF is first taken as the rating embedding model for this thesis's MLT-based solution. Therefore, in domain A, the predicted user-item interaction of NeuMF_DTCDR for user u_i on item v_j can be defined as:

$$
\begin{aligned}
&\hat{y}_{ij} = f(\begin{bmatrix} P_i^a \\ Q_j^a \end{bmatrix}), \\
&P_i^a = \begin{cases} U_i \otimes UC_i, & u_i \in \mathcal{U}^{ac}; \\ U_i^a \otimes UC_i^a, & u_i \in \mathcal{U}^{ad}, \end{cases} \qquad Q_j^a = V_j^a \otimes VC_j^a, \\
&\begin{bmatrix} U_i^a \\ V_j^a \end{bmatrix} \leftarrow h^\top \begin{bmatrix} \phi_{ij}^{GMF^a} \\ \phi_{ij}^{MLP^a} \end{bmatrix}, \qquad h \leftarrow \begin{bmatrix} \alpha h^{GMF} \\ (1-\alpha) h^{MLP} \end{bmatrix},
\end{aligned}
\tag{4.11}
$$

where $f(*)$ is *ReLU* function, h^{GMF} and h^{MLP} denote the edge weights of the GMF and MLP model output layers, respectively, and α is a hyper-parameter. Additionally, $\mathcal{U}^{ac}$ and $\mathcal{U}^{ad}$ represent the common and distinct users, respectively, in domain A. Moreover, $\otimes$ represents a combination operator (i.e., Concat, MP or AP). Here, providing flexibility and determining which operator is more suitable in a special CDR scenario is the aim. Likewise, the predicted user-item interactions of NeuMF_DTCDR can be obtained in domain B.

4.3.3.2 A DTCDR model through DMF (DMF_DTCDR)

DMF was then taken as the rating embedding model for the MLT-based framework. Based on the DMF model, the three different combination operators were also chosen to combine the latent factors and the text vectors for users and items.

In domain A, the predicted user-item interaction of DMF_DTCDR for user u_i on item v_j can be redefined as:

$$
\begin{aligned}
&\hat{y}_{ij} = cosine(P_i^a, Q_j^a) = \frac{[P_i^a]^\top Q_j^a}{\|Q_i^a\| \|Q_j^a\|}, \\
&U_i^a = f(...f(W_{U_2}^a f(r_{i*} W_{U_1}^a))), \\
&V_j^a = f(...f(W_{V_2}^a f(r_{*j} W_{V_1}^a))),
\end{aligned}
\tag{4.12}
$$

where $f(*)$ is *ReLU* function, r_{i*} represents user u_i's ratings across all items, r_{*j} repre-

Table 4.1: The experimental datasets for DTCDR

Datasets	Douban			MovieLens
Domains	Music	Book	Movie	Movie
#Users	1,672	2,110	2,712	10,000
#Items	5,567	6,777	34,893	9,395
#Interactions	69,709	96,041	1,278,401	1,462,905
Sparsity	99.25%	99.33%	98.65%	98.44%

sents item v_j's ratings across all users, W_{U_1}, W_{U_2}... and W_{V_1}, W_{V_2}... are the weights of the multilayer networks in the different layers for U and V, respectively. Additionally, P_i^a and Q_j^a have been formulated in Equation (4.11). Likewise, the predicted user-item interactions of DMF_DTCDR can be obtained in domain B.

4.4 Experiments on DTCDR

Extensive experiments were conducted on real-world datasets to answer the following five key questions:

- **Q1**: How does this thesis's approach outperform the state-of-the-art single-domain and cross-domain models? (See Result 1)

- **Q2**: How does the dimension k of latent factors and text vectors affect the performance of this thesis's models? (See Result 2)

- **Q3**: How do the three combination operators of MTL affect the performance of this thesis's models? (See Result 3)

- **Q4**: How do document embedding and MTL contribute to performance improvement? (See Result 4)?

- **Q5**: How does this thesis's approach perform on Top-N recommended lists? (See Result 5)

4.4.1 Experimental Settings

4.4.1.1 Datasets

Four real-world datasets were chosen for the experiments: the benchmark dataset MovieLens 20M [35] and three Douban datasets (including DoubanMusic, DoubanBook and DoubanMovie, which were crawled from the Douban website). The three Douban datasets were filtered, and the users and items with at least 5 interactions were kept. Additionally, for the MovieLens 20M dataset, $10,000$ users who have at least five interactions were chosen. The details are listed in Table 4.1.

The three Douban datasets contained user profiles, item details, ratings, reviews and tags, while the MovieLens dataset contained item genres, ratings and tags. Three experimental tasks were designed based on these four datasets, as described in the following section.

4.4.1.2 Experimental Tasks

To validate the performance of the DTCDR models and baseline models in different CDR scenarios, two CDR tasks were designed (Tasks 1 and 2). In addition, as mentioned in Section 4.3, the DTCDR models can apply to CSR scenarios, in which there are common items only. Task 3 was thus designed to validate the performance of the DTCDR framework and baseline models in a CSR scenario. The detailed tasks are listed as follows:

- **Task 1**: DoubanMovie+DoubanBook ($2,106$ common users).
- **Task 2**: DoubanMovie+DoubanMusic ($1,666$ common users).
- **Task 3**: DoubanMovie+MovieLens ($4,115$ common movies).

4.4.1.3 Parameter Setting

For a fair comparison, both the parameters of the DTCDR models and those of the baseline models were optimised. For the *Input Layer* and *Embedding Layer* in Figure

4.2, the hyper-parameters of the Doc2Vec model were set, as suggested in [56], and the dimension k of the text vectors and latent factors as $\{8, 16, 32, 64\}$. In the *Neural Network Layer*, the structure is $e \to 32 \to 16 \to k$, where k is the output size (i.e., the dimension of the latent factors) and e is the combined size. For different rating embedding models and combination operators, e has different values. For example, if the combination operator is Concat for DMF_DTCDR, then $e = 2 * k$; elsewise, $e = k$. The parameters of the neural network are initialised as the Gaussian distribution $X \sim N(0, 0.01)$. For NeuMF_DTCDR, $\lambda = 0.001$, the learning rate is 0.001 and the batch size is $1,024$, while for DMF_DTCDR, λ was set as 0.001, the learning rate as 0.0001 and the batch size as 256. The Adaptive Moment Estimation (Adam) [50] was in this thesis's models. Additionally, the number of training iterations num_iter was set as 50 and the best performance was reported in the experimental results.

Table 4.2: The comparison of the baselines and our models (DTCDR)

Model			Training data	Embedding strategy	Transfer strategy (for single-target CDR or dual-target CDR)
Baselines	Single-domain recommendation (SDR)	BPR [91]	Rating	MF	-
		NeuMF [38]	Rating	MLP	-
		DMF [123]	Rating	MLP	-
	Single-target cross-domain recommendation (CDR)	CTR-RBF [122]	Rating & Content	MF	Transfer learning
		BPR_EMCDR_LIN [71]	Rating	MF	Linear matrix translation
		BPR_EMCDR_MLP [71]	Rating	MF	MLP
		BPR_DCDCSR [142]	Rating	MF	Feature combination & MLP
Our DTCDR models	NeuMF-based	NeuMF_DTCDR_Concat	Rating & Content	MLP	MTL & **Concatenation**
		NeuMF_DTCDR_MP	Rating & Content	MLP	MTL & **Max-pooling**
		NeuMF_DTCDR_AP	Rating & Content	MLP	MTL & **Average-pooling**
	DMF-based	DMF_DTCDR_Concat	Rating & Content	MLP	MTL & **Concatenation**
		DMF_DTCDR_MP	Rating & Content	MLP	MTL & **Max-pooling**
		DMF_DTCDR_AP	Rating & Content	MLP	MTL & **Average-pooling**

4.4.1.4 Evaluation Metrics

The ranking-based evaluation strategy was adopted in this study (i.e., the *leave-one-out evaluation*), which has been widely used in the baseline models (e.g., BPR, NeuMF and DMF). That is, for each test rating from a test user on a test item, 99 unrated items were randomly sampled for the test user, and then the test item was ranked among the 100 items. The recommendation performance is evaluated by two metrics — the *Hit Ratio (HR)* and the *Normalised Discounted Cumulative Gain (NDCG)* [38], *HR* measures whether the test item is ranked on the top-N list, while *NDCG* measures the specific ranking quality that assigns high scores to hits at top position ranks.

4.4.1.5 Comparison Methods

The NeuMF_DTCDR and DMF_DTCDR models were compared with the following seven baseline models in two groups of SDR and CDR, respectively. These baseline models are the most relevant methods because each model is a representative or state-of-the-art method with different embedding and transfer strategies. For a clear comparison, the detailed training data types, embedding strategies and transfer strategies of the baseline models and DTCDR models are listed in Table 4.2.

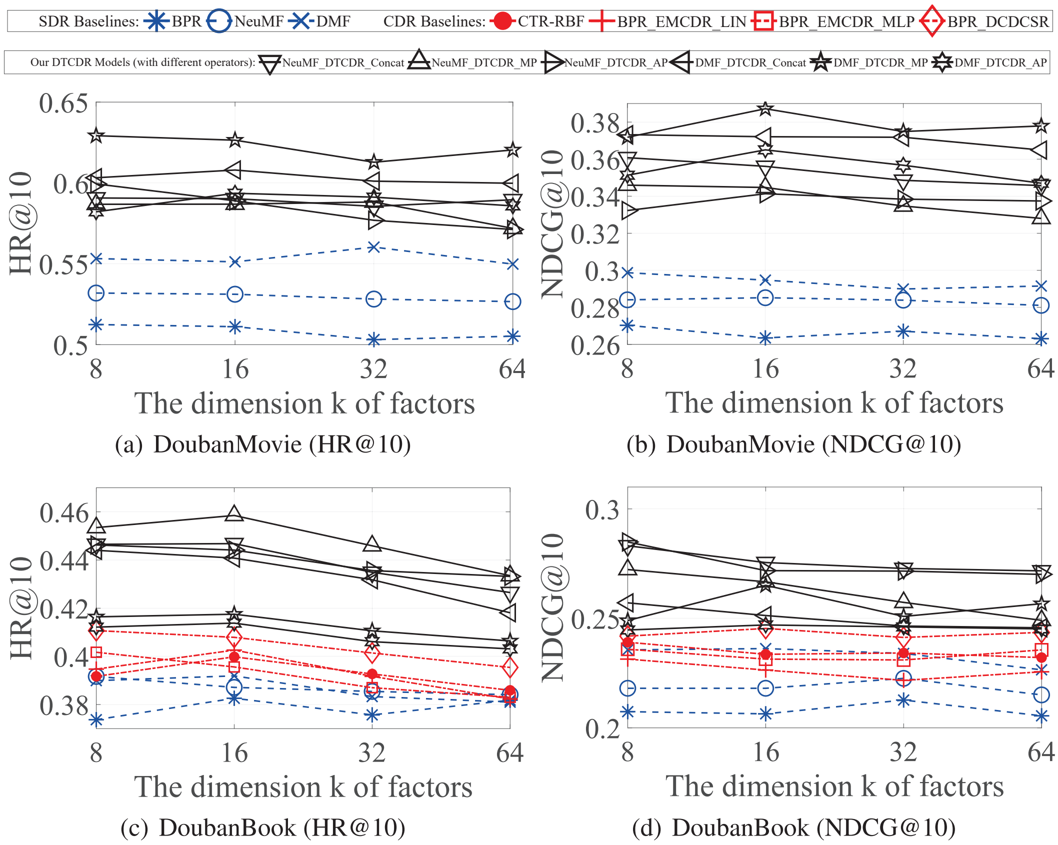
(1) Single-domain recommendation (SDR baselines)

- **BPR** [91] is a representative pairwise learning-based MF model that focuses on minimising the ranking loss between predicted ratings and observed ratings.

- **NeuMF** [38] is a representative NN-based CF model that replaces the conventional inner product with a neural architecture to improve recommendation accuracy.

- **DMF** [123] is a state-of-the-art NN-based CF model that employs a deep architecture to learn the low-dimensional factors of users and items.

(2) Cross-domain recommendation (CDR baselines)



(a) DoubanMovie (HR@10) (b) DoubanMovie (NDCG@10)

(c) DoubanBook (HR@10) (d) DoubanBook (NDCG@10)

Figure 4.3: The experimental result of Task 1. Note: DoubanBook is the target domain for CDR baseline models

- *A non-linear transfer-learning framework* (**CTR-RBF**) [122] is a framework that incorporates the review text. This is a state-of-the-art CDR model, considering both content and rating information for generating user and item embeddings.

- The *EMCDR* [71] utilises LIN and an MLP to represent the relations between the latent factors of two domains. The most promising models — **BPR_EMCDR_LIN** and **BPR_EMCDR_MLP** — were implemented into this framework as baselines.

- The *DCDCSR framework* [142] is a state-of-the-art deep framework that transfers latent factors across domains. The most promising model — **BPR_DCDCSR**

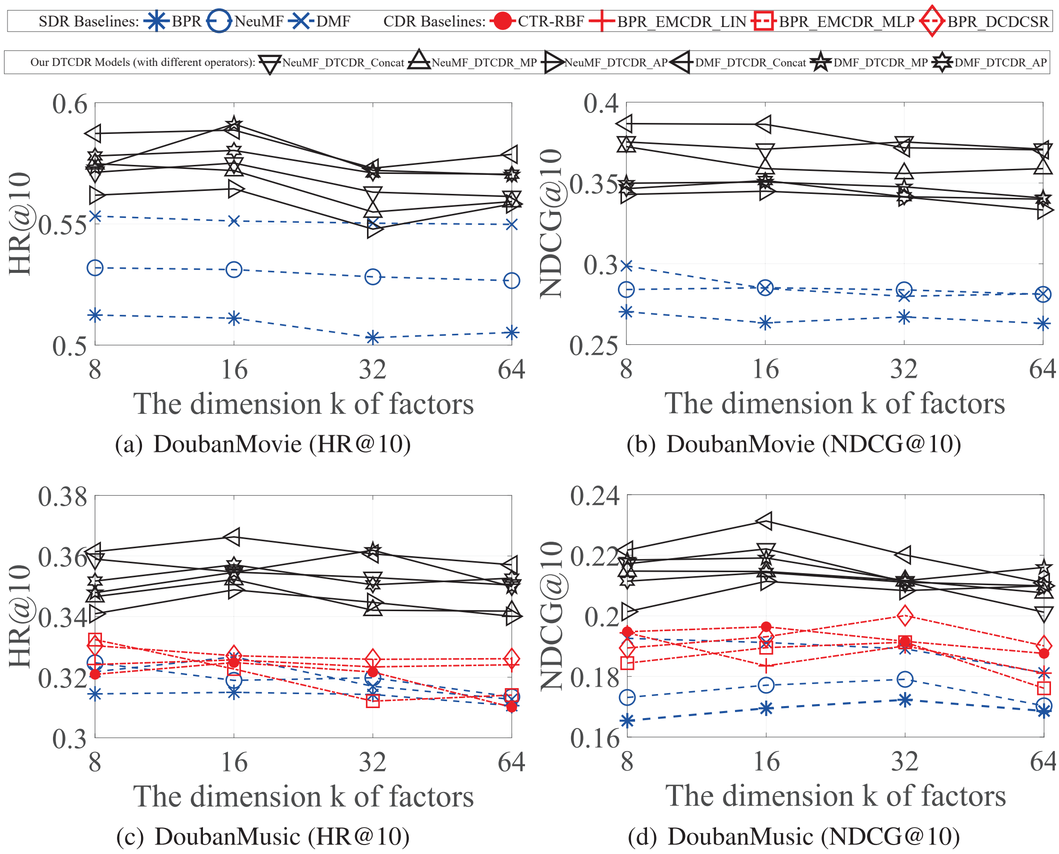


Figure 4.4: The experimental result of Task 2. Note: DoubanMusic is the target domain for CDR baseline models

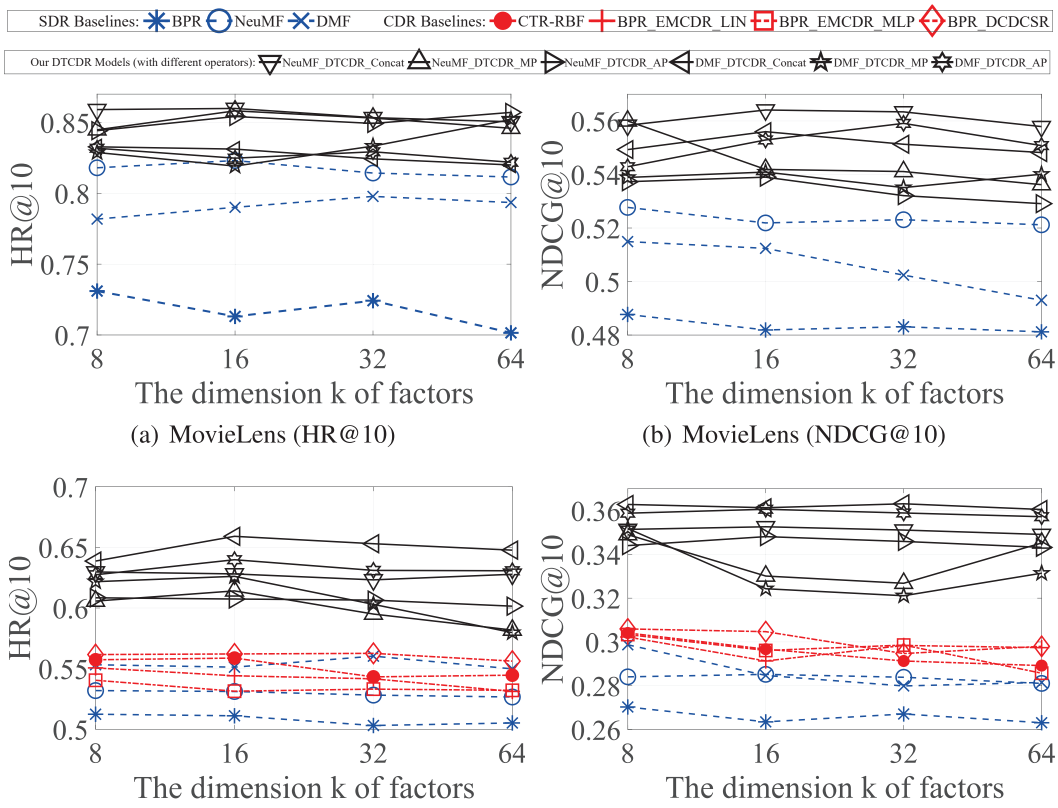
— was implemented as a baseline in the experiments.

4.4.2 Performance Comparison and Analysis

To answer the five key questions **Q1-Q5**, the following experiments were conducted and the corresponding results were analysed.

4.4.2.1 Result 1: Performance Comparison (for Q1)

To answer **Q1**, the performance of the proposed NeuMF_DTCDR and DMF_DTCDR models are compared with the performance of the seven baseline models. For the three SDR models, they were trained in both domains, and the corresponding experimental



(a) MovieLens (HR@10)
(b) MovieLens (NDCG@10)
(c) DoubanMovie (HR@10)
(d) DoubanMovie (NDCG@10)

Figure 4.5: The experimental result of Task 3. Note: DoubanMovie is the target domain for CDR baseline models

results were obtained. For the four cross-domain baseline models, they were trained based in the two domains and then validated in the target (sparser) domain. The target domains were DoubanBook, DoubanMusic and DoubanMovie for Tasks 1, 2 and 3, respectively.

Figures 4.3, 4.4 and 4.5 show the performance of HR@10 and NDCG@10 with different factor dimensions for Tasks 1, 2 and 3, respectively. In both the sparser and richer domains, the NeuMF
_DTCDR and DMF_DTCDR models, on average, outperformed the three single-domain baseline models (BPR, NeuMF and DMF) by 13.73%, 10.82% and 8.97% respectively for HR@10, and by 18.83%, 13.98%, and 11.77% respectively for NDCG@10. In the sparser (target) domains, the NeuMF_DTCDR and DMF_DTCDR models, on average,

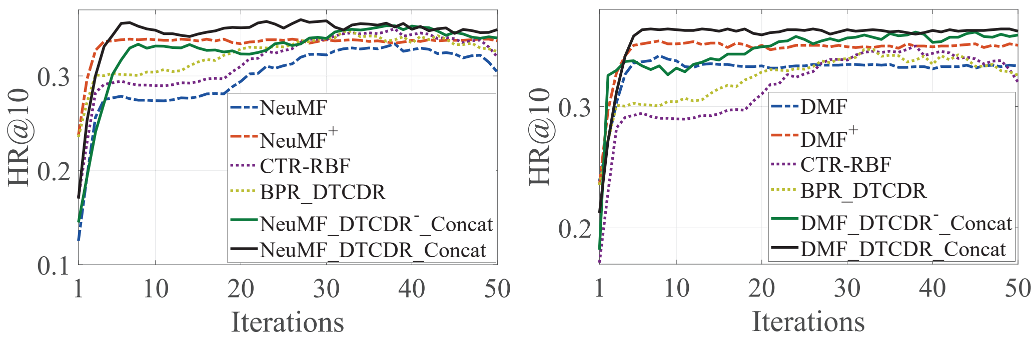


Figure 4.6: Performance comparison with and without document embedding (DE) on Douban-Music ($k = 8$ and the combination operator is Concat). Note that **NeuMF$^+$** and **DMF$^+$** represent NeuMF and DMF with DE while **NeuMF_DTCDR$^-$** and **DMF_DTCDR$^-$** represent NeuMF_DTCDR and DMF_DTCDR without DE

outperformed the four cross-domain baseline models (CTR-RBF, BPR_EMCDR_LIN, BPR_EMCDR_MLP and BPR_DCDCSR) by 9.63%, 9.90%, 10.06% and 6.20% respectively for HR@10, and by 11.91%, 13.40%, 13.84% and 9.37% respectively for NDCG@10. It is worth noting that in most cases (46 out of 48 cases), the worst-performing model still outperformed the best-performing baseline model on average by 3.09% for HR@10 and by 5.88% for NDCG@10. In all cases, this thesis's best-performing model improved the best-performing baseline model on average by 9.45% for HR@10 and by 13.90% for NDCG@10.

Summary 1: In general, the NeuMF_DTCDR and DMF_DTCDR models outperformed both the single-domain baseline models and the cross-domain baseline models. This is because this thesis's models could leverage the richness and diversity of the information of both domains, as well as effectively share the embeddings of common users across domains and avoid over-fitting. These models can improve the recommendation performance in the two domains or systems, which illustrates the effectiveness of the DTCDR models.

4.4.2.2 Result 2: The Effect of Factor Dimension (for Q2)

To answer **Q2**, the effect of k on model performance is analysed in Figures 4.3, 4.4 and 4.5. Specifically, when $k \in \{8, 16\}$, this thesis's models can achieve the best performance on the three experimental tasks. When $k \in \{32, 64\}$, the performance of the models begins to gradually decline. This is because the number of neural network parameters geometrically increases with k, while the training data is relatively sparser; this can lead to over-fitting when $k > 16$.

Summary 2: The dimension k of factors is a sensitive parameter in the NeuMF_DTCDR and DMF_DTCDR models. In general, when $k \leq 16$, the recommendation performance of the NeuMF_DTCDR and DMF_DTCDR models increased with k. However, when $k > 16$, the performance began to gradually decline due to over-fitting.

4.4.2.3 Result 3: The Effect of Combination Operators (for Q3)

To answer **Q3**, the performance of the NeuMF_DTCDR and DMF_DTCDR models was compared, with both models having different combination operators (Concat, MP and AP). As depicted in Figures 4.3, 4.4 and 4.5, when compared to *MP* and *AP*, it could be observed that the *Concat*, NeuMF_DTCDR and DMF_DTCDR models achieved the best performance in most cases. This is because, to a large extent, *Concat* can preserve all embeddings of different domains. *MP* tends to preserve dominating factors and lose generality, while *AP* preserves generality but is easily affected by noisy embeddings.

Summary 3: *Concat* can cause a better and more stable performance for the NeuMF_DTCDR and DMF_DTCDR models. In some isolated cases, *MP* can also achieve effective performance. The performance of *AP* is worse than the two operators, mainly because *AP* can be easily affected by noisy embeddings.

4.4.2.4 Result 4: Contributions of Document Embedding and Multi-Task Learning (for Q4)

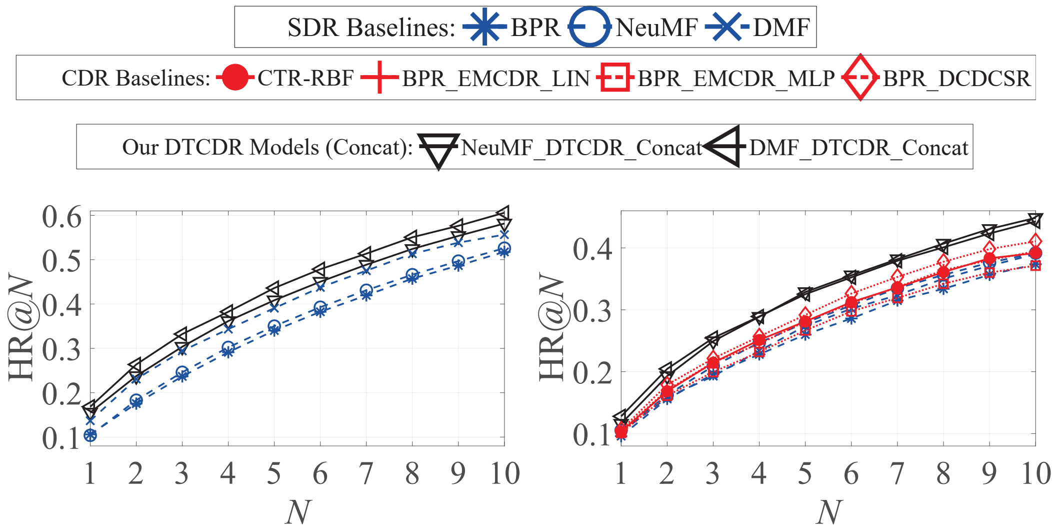
The performances of NeuMF, DMF, NeuMF_DTCDR, DMF_DTCDR (with and without document embedding), CTR-RBF and BPR_DCDCSR were compared. This experiment as conducted on the DoubanMusic dataset and was evaluated by HR@10 with $k = 8$. As can be observed from Figure 4.6, with the assistance of document embedding, the performance of NeuMF$^+$ and DMF$^+$ is always better than their pure models (NeuMF and DMF). According to the best performance among all 50 iterations, their improvements were 1.50% and 2.34%, respectively. Meanwhile, NeuMF_DTCDR and DMF_DTCDR outperformed their simplified models (without document embedding) (NeuMF_DTCDR$^-$ and DMF_DTCDR$^-$) by 1.84% and 1.07%, respectively.

Additionally, without document embedding and only based on MTL, the NeuMF_DTCDR$^-$ still outperformed NeuMF, NeuMF$^+$, CTR-RBF and BPR_DCDCSR after 27 iterations. Similarly, the DMF_DTCDR$^-$ still outperformed DMF, DMF$^+$, CTR-RBF and BPR_DCDCSR after 20 iterations. According to the recorded best performance of the models among all 50 iterations, this thesis's models, NeuMF_DTCDR$^-$ and DMF_DTCDR$^-$, outperformed all six baseline models (NeuMF, NeuMF$^+$, DMF, DMF$^+$, CTR-RBF and BPR_DCDCSR) by 6.12%, 4.79%, 5.72%, 4.38%, 3.35% and 2.07%, respectively.

Summary 4: Document embedding can improve recommendation performance, as the text vectors that were learned by document embedding can provide more prior knowledge to the recommendation models instead of only the initialisation by a random or Gaussian distribution. Additionally, without document embedding, the NeuMF-DTCDR$^-$ and DMF-DTCDR$^-$ can still achieve good performance, so long as the models can be well trained. This is because the MTL technique is applied for sharing the features of common users and items across domains, which can effectively mitigate the data sparsity problem.

4.4.2.5 Result 5: Performance of Top-N Recommendation (for Q5)

To answer **Q5**, the performance of Top-N recommendation is compared in terms of HR@N, in which the ranking position N ranges from 1 to 10 and k is 8. To clearly demonstrate the comparison of the performance, only the performances of NeuMF_DTCDR_Concat, DMF_DTCDR_Concat and the baseline models were report-



(a) Douban-Movie (HR@10) (b) Douban-Book (HR@10)

Figure 4.7: The result of Top-N recommendation for Task 1

ed for Task 1. As can be observed from Figure 5.4, the performance of the NeuMF_DTCDR_Concat and DMF_DTCDR_Concat is consistently better in both datasets than that of the other seven baseline models. In regard to the Douban-Movie, DMF_DTCDR_Concat depicted a better performance than NeuMF_DTCDR_Concat, while for the Douban-Book, the performance of NeuMF_DTCDR_Concat was better than that of DMF_DTCDR_Concat when N was greater than four. In this case, the NeuMF_DTCDR_Concat and DMF_DTCDR_Concat improved, on average, the single-domain baselines (BPR, NeuMF and DMF) by 18.36%, 15.62% and 12.24% respectively. They also improved the cross-domain baselines (CTR-RBF, BPR_EMCDR_LIN, BPR_EMCDR_MLP and BPR_DCDCSR) by 12.16%, 12.20%, 16.71% and 8.55%.

Summary 5: In general, the NeuMF_DTCDR_Concat and DMF_DTCDR_Concat outperformed the seven baseline models for Top-N recommendation, and DMF_DTCDR_Concat demonstrated a better performance than NeuMF_DTCDR_Concat in most cases. This is because this thesis's models can leverage the richness and diversity of the information of both domains, as well as effectively share the embeddings of common users across domains.

4.5 Summary

In this chapter, a general framework for Dual-Target Cross-Domain Recommendation, called DTCDR, has been proposed, which leverages ratings and multi-source content to improve the recommendation performance on dual-target domains simultaneously. The document embedding and rating embedding techniques are proposed to generate the text and rating embeddings of users and items. Based on multi-task learning, a flexible and effective embedding-sharing strategy is adopted to combine and share the embeddings of common users across domains. Finally, extensive experiments conducted on real-world datasets have demonstrated the superior performance of our models.

Chapter 5

A Graphical and Attentional Framework for Dual-Target Cross-Domain Recommendation

Existing CDR approaches tend to leverage the auxiliary information on a richer domain to only help improve the recommendation accuracy on a sparser domain, which results in single-target CDR. In contrast, the novel dual-target CDR has been recently proposed to improve the recommendation accuracies in both richer and sparser domains simultaneously by effectively utilising the information or knowledge from both domains [141, 63].

Inspired by the DTCDR framework proposed in Chapter 4, this chapter attempts to achieve a higher goal — further improving the recommendation performance in both domains. However, in addition to the original challenges **CH2** and **CH3** that have been introduced in Sections 1.2.2 and 1.2.3, the higher goal faces a new challenge that has introduced in Section 1.2.4 and expressed by **CH4**: *'How can the user or item embeddings in each target domain be effectively optimised for improving recommendation accuracies in both domains?'*.

In this chapter, the dual-target CDR problem is first normalised. Then, to address the challenges **CH2**-**CH4**, the novel GA-DTCDR is proposed, followed by its detailed components. This detailed framework is explained in the following sections.

5.1 Problem Statement

First, for readability purposes, the important notations section is listed in Table A.3. Dual-target CDR is defined as follows:

Definition 5. Dual-target cross-domain recommendation

- **Input**: Given two related domains a and b including explicit feedback (e.g., ratings and comments), implicit feedback (e.g., purchase and browsing histories) and side information (e.g., user profiles and item details).

- **Output**: The DTCDR aims to improve the recommendation accuracies in both domains simultaneously by leveraging their observed information.

Note that a certain degree of overlap between the users of domains a and b (i.e., common users) plays a key role in bridging the two domains and in exchanging knowledge across domains. This is a common idea in the existing single-target and dual-target CDR approaches.

5.2 The Proposed GA-DTCDR

The novel GA-DTCDR is proposed for targeting the dual-target CDR problem. As shown in Figure 5.1, this framework is divided into five main components: the *Input Layer*, *Graph Embedding Layer*, *Feature Combination Layer*, *Neural Network Layers* and *Output Layer*. The details of each component will be presented below.

5.2.1 Input Layer

First, for the input of the GA-DTCDR, both explicit feedback (ratings and comments) and side information (user profiles and item details) were considered. These input data can be generally classified into two categories: rating information and content information.

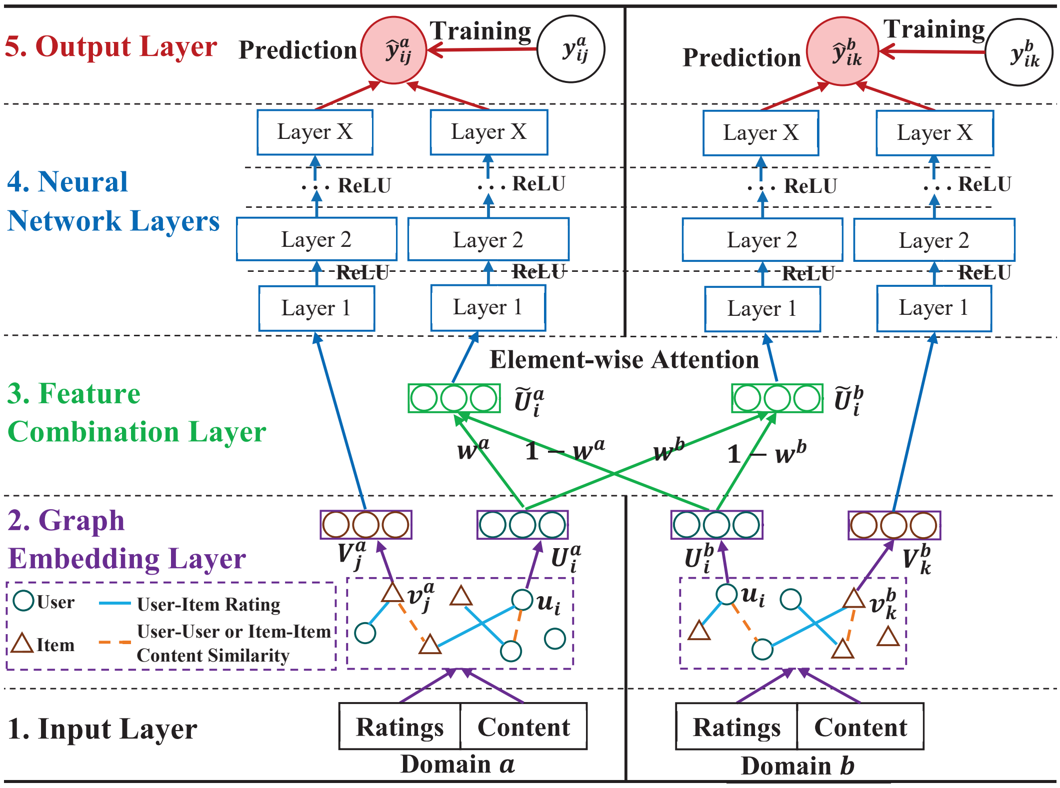


Figure 5.1: The structure of the GA-DTCDR framework

5.2.2 Graph Embedding Layer

The rating and content information of domains a and b were then leveraged to construct a heterogeneous graph that represents user-item interaction relationships, user-user similarity relationships and item-item similarity relationships. Based on the graph, the graph-embedding model, Node2vec [33], was applied to generate user and item embedding matrices.

5.2.2.1 Feature Combination Layer

An element-wise attention mechanism was then proposed to combine the common users' embeddings for domains a and b. This layer intelligently provides a set of weights to the two embeddings of a common user that were learned from both domains; it also generates a combined embedding for the common user, which remains

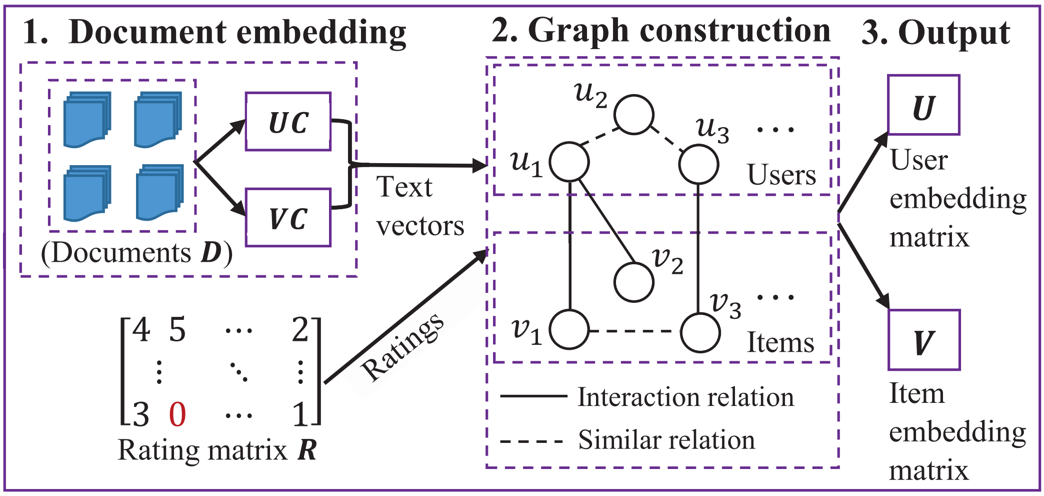


Figure 5.2: Graph embedding

the user's features that were learned from domains a and b with different proportions.

5.2.3 Neural Network Layers

In this component, a fully connected neural network (i.e., MLP) is applied to represent a non-linear relationship between users and items in each domain.

5.2.4 Output Layer

Last, final user-item interaction predictions can be generated. The training of this model is mainly based on the loss between predicted user-item interactions and observed user-item interactions.

In fact, like the single-target or dual-target CDR approaches in [139, 142, 141], the GA-DTCDR framework can also be applied to CSR, in which the two systems have the same domain but different users and thus contain common items only, such as DoubanMovie and MovieLens (see Task 3 in **Experiments and Analysis**). Accordingly, as shown in Figure 5.1, common users need only be replaced with common items for supporting dual-target CSR.

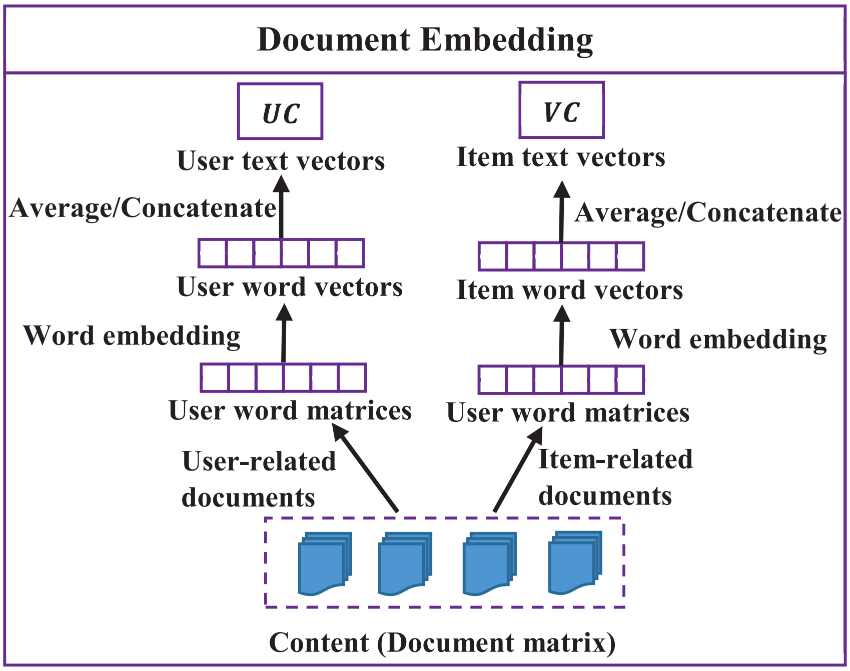


Figure 5.3: Document embedding

5.3 Graph Embedding Layer

Existing embedding strategies for RSs mainly focus on representing the user-item interaction relationship. Apart from this relationship, a graph is used to also represent user-user and item-item relationships. Therefore, based on the rating and content information that was observed from domains a and b, two heterogeneous graphs were constructed, including nodes (users and items) and weighted edges (ratings and content similarities), for domains a and b, respectively. More representative user and item embedding matrices could then be genereted. The details of graph embedding are presented in Figure 5.2, which contains three main components: *Document Embedding*, *Graph Construction* and *Output*.

5.3.1 Document Embedding

To construct the heterogeneous graph, the content similarities between two users or two items must be computed. To this end, multi-source content information is considered (e.g., reviews, tags, user profiles, item details), which was observed from domains a and b, to generate user and item content embedding matrices. In this chapter, the most widely used model is adopted (i.e., Doc2vec [56]) as the document embedding technique. The detailed document embedding process worked as follows: 1) In the training set, for a user u_i, the comments (reviews and tags) C_{i*} and the user profile up_i of u_i were collected in the same document d_i, while for an item v_j, the comments (reviews and tags) C_{*j} on the item and its item detail id_j were collected in the same document d_{m+j}; 2) the words were then segmented in the documents $D = \{d_1, d_2, ..., d_{m+n}\}$ by using the most widely used natural language tool — StanfordCoreNLP [72]; 3) Finally, the Doc2vec model was applied to map the documents D into the text vectors UC and VC for users and items, respectively. Document Embedding details are presented in Figure 5.3.

5.3.1.1 Graph Construction

The users and items are first linked through their interaction relationships. The weights of these interaction edges are normalised ratings (i.e., $R/max(R)$). To consider the user-user and item-item relationships in the heterogeneous graph, the synthetic edges were generated between two users or two items according to their normalised content similarities (edge weights). The generation probability $P(i, l)$ of the edge between users u_i and u_l is expressed as follows:

$$P(i, l) = \alpha \cdot sim(UC_i, UC_l), \tag{5.1}$$

where α is a hyper-parameter that controls the sampling probability and $sim(UC_i, UC_l)$ is the normalised cosine similarity between UC_i and UC_l. Similarly, the generation probability between two items can be obtained. Based on the user-item interaction relationships, user-user similarity relationships and item-item similarity relationships, the heterogeneous graphs G^a and G^b can be constructed for domains a and b, respectively.

5.3.1.2 The Output of Document Embedding

Based on the two heterogeneous graphs G^a and G^b, the graph-embedding model Node2vec [33] is employed to generate a user embedding matrix U and item embedding matrix V.

5.4 Feature Combination Layer

In this layer, the embeddings of common users learned from domains a and b are combined by an element-wise attention mechanism. In this way, the combined embeddings of common users $\tilde{U}$ for each domain can remain both features that were learned from the two domains in different proportions. The traditional attention mechanism tends to selectively focus on a certain part of representative features, and it provides these features higher weights when generating the combined features [5]. Similarly, for a common user u_i, the element-wise attention mechanism tends to pay more attention to the more informative elements from each pair of elements in U_i^a and U_i^b, respectively. The element-wise attention mechanism can thus generate two more representative embeddings $\tilde{U}_i^a$ and $\tilde{U}_i^b$ of the common user u_i for domains a and b, respectively.

Recalling the motivating example in this thesis's Introduction, the GA-DTCDR can pay more attention to Alice's movie features, while the approach can pay more attention to Bob's book features. Based on this idea, the GA-DTCDR thus combines his/her movie and book features to improve the recommendation accuracies in both movie and book domains simultaneously.

The structure of element-wise attention is demonstrated in the *Feature Combination Layer* of Figure 4.2. The combined embedding $\tilde{U}_i^a$ of a common user u_i for domain a can be represented as:

$$\tilde{U}_i^a = W^a \odot U_i^a + (1 - W^a) \odot U_i^b, \tag{5.2}$$

where $\odot$ is the element-wise multiplication and $W^a \in \mathbb{R}^{m^a \times k}$ is the weight matrix for the attention network in domain a. Similarly, the combined embedding $\tilde{U}_i^b$ of u_i can be obtained for domain b.

Note that for distinct users and all the items in domains a and b, their embeddings are reserved without using the attention mechanism. This is because they do not have dual embeddings in both domains a and b.

5.5 Training for Neural Network Layers and Output Layer

First, the model is trained with the following objective function in domain a:

$$\min_{P^a, Q^a, \Theta^a} \sum_{y \in Y^{a+} \cup Y^{a-}} \ell(y, \hat{y}) + \lambda(\|P^a\|_F^2 + \|Q^a\|_F^2), \tag{5.3}$$

where $\ell(y, \hat{y})$ is a loss function between an observed interaction y and its corresponding predicted interaction $\hat{y}$ (see Eq. (5.5)), Y^{a+} and Y^{a-} denote all the observed and unobserved user-item interactions in domain a respectively, $\|P^a\|_F^2 + \|Q^a\|_F^2$ is the regulariser (see Eq. (5.6)) and λ is a hyper-parameter that controls the importance of the regulariser. To avoid this model becoming over-fitted to Y^+ (positive instances), a certain number of unobserved user-item interactions were randomly selected as negative instances, denoted by $Y^-_{sampled}$, to replace Y^-. This training strategy has been widely used in existing approaches [38].

Based on rating information, the user-item interaction y_{ij} between a user u_i and an item v_i can be represented as:

$$y_{ij} = \begin{cases} r_{ij}, & \text{if } y_{ij} \in Y^+; \\ 0, & \text{if } y_{ij} \in Y^-_{sampled}; \\ null, & \text{otherwise.} \end{cases} \tag{5.4}$$

A normalised cross-entropy loss was chosen, which can be represented as:

$$\ell(y, \hat{y}) = \frac{y}{max(R)} \log \hat{y} + (1 - \frac{y}{max(R)}) \log(1 - \hat{y}), \tag{5.5}$$

where $max(R)$ is the maximum rating in a domain.

As shown in the *Neural Network Layers* of Figure 5.1, the GA-DTCDR framework employed a neural network (i.e., MLP) to represent a non-linear relationship between users and items. The input-embedding matrices of users and items in domain a for the MLP are $P^a_{in} = [\tilde{U}^a; U^{ad}]$ and $Q^a_{in} = V^a$ respectively, where $\tilde{U}^a$ is the combined embedding matrix of common users for domain a and U^{ad} is the embedding matrix of distinct users in domain a. Therefore, the embedding of user u_i and item embedding of item v_j in the output layer of the MLP can be represented as:

$$\begin{aligned} P^a_i &= P^a_{out_i} = f(...f(f(P^a_{in_i} \cdot W^a_{P_1}) \cdot W^a_{P_2})), \\ Q^a_j &= Q^a_{out_j} = f(...f(f(Q^a_{in_j} \cdot W^a_{Q_1}) \cdot W^a_{Q_2})), \end{aligned} \tag{5.6}$$

where the activation function $f(*)$ is ***ReLU***, $W^a_{P_1}, W^a_{P_2}...$ and $W^a_{Q_1}, W^a_{Q_2}...$ are the weights of multilayer networks in different layers in domain a for $P^a_{in_i}$ and $Q^a_{in_j}$, respectively.

Finally, in the *Output Layer* of Figure 5.1, the predicted interaction $\hat{y}_{ij}$ between u_i and v_j in domain a is expressed as follows:

$$\hat{y}^a_{ij} = cosine(P^a_i, Q^a_j) = \frac{P^a_i \cdot Q^a_j}{\|P^a_i\| \|Q^a_j\|}. \tag{5.7}$$

Table 5.1: The experimental datasets for GA-DTCDR

Datasets	Douban			MovieLens
Domains	Book	Music	Movie	Movie
#Users	2,110	1,672	2,712	10,000
#Items	6,777	5,567	34,893	9,395
#Interactions	96,041	69,709	1,278,401	1,462,905
Density	0.67%	0.75%	1.35%	1.56%

In contrast to the conventional inner product, the greatest advantage of cosine distance for interaction prediction is that it does not need to normalise separately.

Similarly, the predicted interaction $\hat{y}^b_{ij}$ can be obtained in domain b.

5.6 Experiments on GA-DTCDR

Extensive experiments were conducted on four real-world datasets to answer the following key questions:

- **Q1**: How does this thesis's model outperform the state-of-the-art models? (See Result 1)

- **Q2**: How does the element-wise attention mechanism contribute to performance improvement? (See Result 2)

- **Q3**: How does the dimension k of embeddings affect the performance of the model? (See Result 3)

- **Q4**: How does the model perform on Top-N recommended lists? (See Result 4)?

Table 5.2: The experimental tasks for GA-DTCDR

Tasks		**Sparser**	**Richer**	**Overlap**
CDR	**Task 1**	DoubanBook	DoubanMovie	#Common Users = 2,106
	Task 2	DoubanMusic	DoubanMovie	#Common Users = 1,666
CSR	**Task 3**	DoubanMovie	MovieLens	#Common Items = 4,115

5.6.1 Experimental Settings

5.6.1.1 Experimental Datasets and Tasks

To validate the recommendation performance of the GA-DTCDR approach and baseline approaches, four real-world datasets were chosen (see Table 5.1).

For the three Douban subsets, the users and items with at least 5 interactions were kept, while for MovieLens 20M, a subset MovieLens was extracted that contained $10,000$ users who had also experienced at least 5 interactions. This filtering strategy has been widely used in existing approaches [123, 128]. According to the observed data, the three Douban subsets contained ratings, reviews, tags, user profiles and item details, while the MovieLens subset contained ratings, tags and item details. Based on these four datasets, 2 CDR tasks and 1 CSR task were designed (see Table 5.2) to validate the recommendation performance in CDR and CSR scenarios, respectively.

5.6.1.2 Parameter Setting

For fair comparison, the parameters of the GA-DTCDR and those of the baselines were optimised. For the *Graph Embedding Layer* in Figure 5.1, the hyper-parameters of Doc2vec and Node2vec models were set, as suggested in [56, 33], and the sampling probability α was set as 0.05. In the *Neural Network Layers* of Figure 5.1, the structure of the layers is '$k \to 2k \to 4k \to 8k \to 4k \to 2k \to k$' and the parameters of the neural network are initialised as the Gaussian distribution $X \sim \mathcal{N}(0, 0.01)$. To train the GA-DTCDR, 7 negative instances were randomly selected for each observed positive instance into $Y^{-}_{sampled}$, Adam [50] was adopted to train the neural network and the maximum number of training epochs was to 50. The learning rate was 0.001, the regularisation coefficient λ was 0.001 and the batch size was $1,024$. To answer **Q3**, the dimension k of the embedding varied in $\{8, 16, 32, 64, 128\}$. Finally, the best performance in all 50 training epochs were reported as our experimental results.

5.6.1.3 Evaluation Metrics

To evaluate the recommendation performances of the GA-DTCDR and baseline approaches, the ranking-based evaluation strategy was adopted, which has been widely used in the literature [123, 118]. The latest interaction with a test item was chosen as the test interaction for each test user, 99 unobserved interactions for the test user were randomly sampled and then the test item was ranked among the 100 items. The ranking-based strategy, or the *Leave-one-out evaluation*, includes the two main metrics of*HR* and *NDCG* [118]. *HR*@N is the recall rate, while *NDCG*@N measures the specific ranking quality that assigns high scores to hits at top position ranks. Note that only *HR*@10 and *NDCG*@10 results were reported in **Results 1-3**, and only *HR*@N and *NDCG*@N results were reported in **Result 4**.

Table 5.3: The comparison of the baselines and our methods (GA-DTCDR)

Models / Details	Baselines				
	SDR			Single-target CDR	
	NeuMF [38]	**DMF** [123]	**CTR-RBF** [122]	**BPR_DCDCSR** [142]	**TMH** [41]
Training data	Rating	Rating	Rating & content	Rating	Rating & content
Encoding	One-hot	Rating vector	Topic modeling	Random initialisation	One-hot
Embedding	Non-linear MLP	Non-linear MLP	Linear MF	Linear MF	Non-linear MLP
Transfer strategy	-	-	Mapping & Transfer learning	Combination & MLP	Mapping & transfer learning & attention

Models / Details	Dual-target CDR baselines		Our dual-target CDR models	
	DMF_DTCDR_Concat [141]	**DDTCDR** [63]	**GA-DTCDR_Average** (a variant of GA-DTCDR for ablation study)	**GA-DTCDR**
Training data	Rating & content	Rating	Rating & content	Rating & content
Encoding	Rating vector	One-hot & multi-hot	Heterogeneous graph	Heterogeneous graph
Embedding	Non-linear MLP	Non-linear MLP	Graph embedding	Graph embedding
Transfer strategy	Multi-task learning & concatenation	Dual transfer learning	Combination (**average-pooling**)	Combination (**element-wise attention**)

5.6.1.4 Comparison Methods

As shown in Table 5.3, the GA-DTCDR is compared with the seven baseline models in three groups: SDR, single-target CDR and dual-target CDR. All these models are representative and/or state-of-the-art approaches for each group. Additionally, for ablation study, a simplified version of the GA-DTCDR was implemented (the GA-DTCDR_Average), which replaced element-wise attention with a fixed combination strategy (i.e., AP). For a clear comparison, Table 5.3 depicts the detailed training data types, encoding strategies, embedding strategies and transfer strategies of all nine models that were implemented in the experiments.

5.6.2 Performance Comparison and Analysis

To answer the four questions **Q1-Q4**, the following experiments were conducted and the corresponding results were analysed.

5.6.2.1 Result 1: Performance Comparison (for Q1)

To answer **Q1**, the performance of this thesis's GA-DTCDR was compared with the performance of the seven baseline models. Note that the SDR baselines were trained in each domain and had their performance in each domain reported; that the single-target CDR baselines were trained in both domains and had only their performance in the sparser domain reported; and that the dual-target CDR models were trained in both domains and had their performance in each domain reported.

Tables 5.4, 5.5, and 5.6 display the experimental results in terms of HR@10 and NDCG@10, with different embedding dimensions k for Tasks 1, 2 and 3 respectively. As indicated in Tables 5.4, 5.5 and 5.6, this thesis's GA-DTCDR outperformed all SDR, single-target CDR and dual-target CDR baselines by an average improvement of 8.46%. Specifically, this thesis's GA-DTCDR improved the best-performing baselines (with results marked by '*' in Tables 5.4, 5.5 and 5.6) by an average of 10.34% for Task 1, 10.29% for Task 2 and 4.76% for Task 3. This is because this thesis's GA-DTCDR

Table 5.4: The experimental results (HR@10 & NDCG@10) for Tasks 1 (the best-performing baselines with results marked by *)

Task	Domain	SDR baselines				Single-target CDR baselines					
		NeuMF		DMF		CTR-RBF		BPR_DCDCSR		TMH	
		HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
Task1	DoubanBook (sparser)	.3810	.2151	.3841	.2265	.3830	.2217	.3954	.2419	.4199	**.2583***
($k = 8$)	DoubanMovie (richer)	.5266	.2911	.5498	.3114	-	-	-	-	-	-
Task 1	DoubanBook (sparser)	.3833	.2181	.3854	.2356	.3870	.2256	.4014	.2413	.4331	**.2522***
($k = 16$)	DoubanMovie (richer)	.5282	.2939	.5573	.3141	-	-	-	-	-	-
Task 1	DoubanBook (sparser)	.3899	.2182	.3871	.2340	.3956	.2264	.4079	.2436	**.4468***	**.2647***
($k = 32$)	DoubanMovie (richer)	.5411	.2991	.5612	.3254	-	-	-	-	-	-
Task 1	DoubanBook (sparser)	.3908	.2226	.3917	.2362	.4017	.2314	.4107	.2454	**.4504***	**.2768***
($k = 64$)	DoubanMovie (richer)	.5449	.3152	.5632	.3387	-	-	-	-	-	-
Task 1	DoubanBook (sparser)	.4012	.2310	.4046	.2451	.4171	.2532	.4111	.2431	**.4523***	**.2814***
($k = 128$)	DoubanMovie (richer)	.5512	.3301	.5776	.3505	-	-	-	-	-	-

Task	Domain	Dual-target CDR baselines				Dual-target CDR (our)				Improvement (GA-DTCDR vs. best baselines)	
		DMF_DTCDR _Concat		DDTCDR		GA-DTCDR _Average		GA-DTCDR			
		HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
Task1	DoubanBook (sparser)	**.4412***	.2571	.4033	.2257	.4057	.2513	**.4479**	**.2759**	1.52%	6.81%
($k = 8$)	DoubanMovie (richer)	**.6032***	**.3732***	.5612	.3185	.5968	.3546	**.6518**	**.4025**	8.06%	7.85%
Task 1	DoubanBook (sparser)	**.4408***	.2513	.4054	.2292	.4190	.2577	**.4706**	**.2900**	6.76%	14.99%
($k = 16$)	DoubanMovie (richer)	**.6080***	**.3721***	.5750	.3595	.6013	.3596	**.6566**	**.4014**	10.80%	7.87%
Task 1	DoubanBook (sparser)	.4318	.2461	.4180	.2344	.4346	.2610	**.4758**	**.2896**	6.50%	9.41%
($k = 32$)	DoubanMovie (richer)	**.6011***	**.3718***	.5739	.3386	.6374	.3896	**.6747**	**.4187**	12.24%	12.61%
Task 1	DoubanBook (sparser)	.4265	.2452	.4258	.2430	.4423	.2671	**.4882**	**.3026**	8.40%	9.32%
($k = 64$)	DoubanMovie (richer)	**.5998***	**.3649***	.5825	.3553	.6416	.3941	**.6817**	**.4205**	13.65%	15.23%
Task 1	DoubanBook (sparser)	.4317	.2510	.4225	.2439	.4490	.2691	**.4995**	**.3098**	10.44%	10.09%
($k = 128$)	DoubanMovie (richer)	**.5991***	**.3680***	.5863	.3589	.6449	.3981	**.6957**	**.4406**	16.12%	19.73%

effectively leverages the richness and diversity of the information in both domains, and because it intelligently and effectively combines the embeddings of common users.

Summary 1: In all cases, this thesis's GA-DTCDR outperformed both the single-domain and cross-domain baseline models. This is because it leverages the richness and diversity of the information of both domains and intelligently combines the embeddings of common users, thus improving the recommendation accuracies on dual-target domains or systems.

5.6.2.2 Result 2: Ablation Study (for Q2)

To answer **Q2**, a variant of this thesis's GA-DTCDR was implemented — the GA-DTCDR_Average — by replacing element-wise attention with AP. AP is the combination strategy used by the existing dual-target CDR approaches [141], and it provides the weight equally (0.5) to the embeddings of common users that were learned

Table 5.5: The experimental results (HR@10 & NDCG@10) for Tasks 2

Task	Domain	**SDR baselines**				**Single-target CDR baselines**					
		NeuMF		DMF		CTR-RBF		BPR_DCDCSR		TMH	
		HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
Task 2	DoubanMusic (sparser)	.3135	.1703	.3127	.1812	.3227	.1895	.3259	.1894	.3579	.2034
($k = 8$)	DoubanMovie (richer)	.5266	.2911	.5498	.3114	-	-	-	-	-	-
Task 2	DoubanMusic (sparser)	.3190	.1731	.3170	.1891	.3121	.1761	.3261	.1901	.3612	.2137
($k = 16$)	DoubanMovie (richer)	.5282	.2939	.5573	.3141	-	-	-	-	-	-
Task 2	DoubanMusic (sparser)	.3198	.1771	.3218	.1912	.3141	.1844	.3271	.1931	**.3701***	**.2202***
($k = 32$)	DoubanMovie (richer)	.5411	.2991	.5612	.3254	-	-	-	-	-	-
Task 2	DoubanMusic (sparser)	.3242	.1791	.3267	.1926	.3324	.1916	.3304	.2001	**.3882***	**.2323***
($k = 64$)	DoubanMovie (richer)	.5449	.3152	.5632	.3387	-	-	-	-	-	-
Task 2	DoubanMusic (sparser)	.3314	.1810	.3301	.1971	.3412	.1954	.3452	.2074	**.3946***	**.2430***
($k = 128$)	DoubanMovie (richer)	.5512	.3301	.5776	.3505	-	-	-	-	-	-

Task	Domain	**Dual-target CDR baselines**				**Dual-target CDR (our)**				**Improvement (GA-DTCDR vs. best baselines)**	
		DMF_DTCDR_Concat		DDTCDR		GA-DTCDR_Average		**GA-DTCDR**			
		HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
Task 2	DoubanMusic (sparser)	**.3614***	**.2117***	.3302	.1930	.3690	.2109	**.3852**	**.2166**	6.59%	2.31%
($k = 8$)	DoubanMovie (richer)	**.5873***	**.3867***	.5655	.3629	.5987	.3731	**.6470**	**.3983**	10.17%	3.00%
Task 2	DoubanMusic (sparser)	**.3663***	**.2213***	.3451	.2092	.3706	.2037	**.3947**	**.2256**	7.75%	1.94%
($k = 16$)	DoubanMovie (richer)	**.5887***	**.3863***	.5704	.3676	.6058	.3716	**.6426**	**.3950**	9.16%	2.25%
Task 2	DoubanMusic (sparser)	.3607	.2201	.3463	.2050	.3789	.2056	**.4133**	**.2318**	14.58%	5.32%
($k = 32$)	DoubanMovie (richer)	**.5770***	**.3758***	.5739	.3726	.6145	.3754	**.6677**	**.4141**	15.72%	10.19%
Task 2	DoubanMusic (sparser)	.3571	.2109	.3466	.2045	.3812	.2144	**.4384**	**.2489**	12.93%	7.15%
($k = 64$)	DoubanMovie (richer)	**.5787***	**.3705***	.5719	.3621	.6120	.3681	**.6817**	**.4284**	17.80%	15.63%
Task 2	DoubanMusic (sparser)	.3580	.2132	.3520	.2117	.3996	.2207	**.4491**	**.2604**	13.81%	7.16%
($k = 128$)	DoubanMovie (richer)	**.5792***	.3742	.5748	**.3762***	.6311	.3859	**.7068**	**.4526**	22.03%	20.31%

from dual domains. As revealed in Tables 5.4, 5.5 and 5.6 with the element-wise attention, this thesis's GA-DTCDR improved GA-DTCDR_Average by an average of 6.76%. This means that element-wise attention plays a crucial role in this thesis's GA-DTCDR and that the existing fixed combination strategies can hardly achieve an effective embedding optimisation in each target domain.

Summary 2: Compared with the GA-DTCDR_Average variant, this thesis's GA-DTCDR can significantly improve its recommendation accuracy. This result indicates that element-wise attention significantly contributes to this thesis's framework.

5.6.2.3 Result 3: Impact of Embedding Dimension k (for Q3)

To answer **Q3**, the effect of k on the performance of the GA-DTCDR framework as depicted in Tables 5.4, 5.5 and 5.6 was analysed. In terms of HR@10 and NDCG@10, the recommendation accuracy of the GA-DTCDR generally increased by k because a

Table 5.6: The experimental results (HR@10 & NDCG@10) for Tasks 3

Task	Domain	SDR baselines				Single-target CDR baselines					
		NeuMF		DMF		CTR-RBF		BPR_DCDCSR		TMH	
		HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
Task 3 ($k=8$)	DoubanMovie (sparser)	.5266	.2911	.5498	.3114	.5514	.3156	.5762	.3347	.5987	.3487
	MovieLens (richer)	.7818	.5024	.8115	.5219	-	-	-	-	-	-
Task 3 ($k=16$)	DoubanMovie (sparser)	.5282	.2939	.5573	.3141	.5631	.3213	.5816	.3438	.6031	.3580
	MovieLens (richer)	.7901	.5084	.8143	.5212	-	-	-	-	-	-
Task 3 ($k=32$)	DoubanMovie (sparser)	.5411	.2991	.5612	.3254	.5721	.3347	.5821	.3447	.6108	**.3733***
	MovieLens (richer)	.7978	.5124	.8180	**.5231***	-	-	-	-	-	-
Task 3 ($k=64$)	DoubanMovie (sparser)	.5449	.3152	.5632	.3387	.5704	.3327	.5926	.3559	.6186	**.3754***
	MovieLens (richer)	.7935	.5149	**.8231***	.5277	-	-	-	-	-	-
Task 3 ($k=128$)	DoubanMovie (sparser)	.5512	.3301	.5776	.3505	.5912	.3741	.6142	.3904	.6314	**.3927***
	MovieLens (richer)	.8042	.5205	**.8319***	.5344	-	-	-	-	-	-

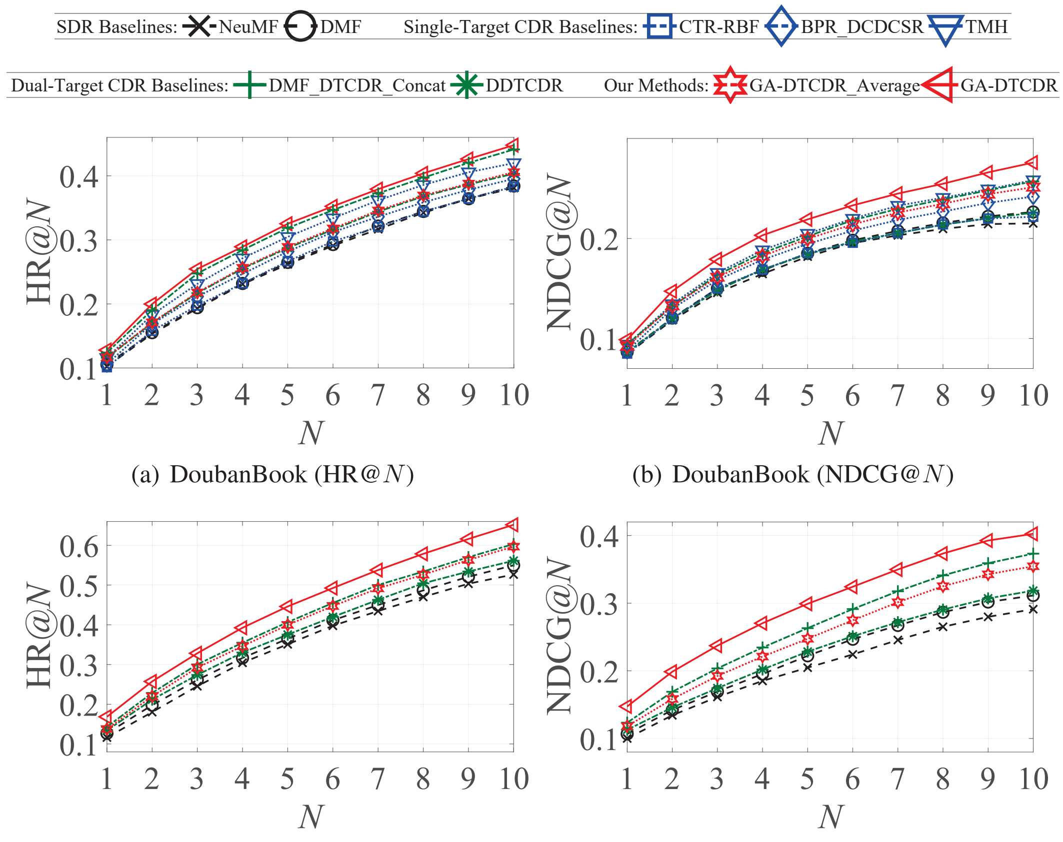
Task	Domain	Dual-target CDR baselines				Dual-target CDR (our)				Improvement (GA-DTCDR vs. best baselines)	
		DMF_DTCDR _Concat		DDTCDR		GA-DTCDR _Average		GA-DTCDR			
		HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG	HR	NDCG
Task 3 ($k=8$)	DoubanMovie (sparser)	**.6387***	**.3628***	.6070	.3522	.6140	.3572	**.6486**	**.4005**	6.85%	10.39%
	MovieLens (richer)	**.8328***	**.5293***	.8211	.5283	.8225	.5241	**.8541**	**.5372**	2.56%	1.49%
Task 3 ($k=16$)	DoubanMovie (sparser)	**.6391***	**.3606***	.6100	.3518	.6266	.3710	**.6514**	**.4018**	1.92%	11.43%
	MovieLens (richer)	**.8312***	**.5260***	.8263	.5170	.8280	.5277	**.8547**	**.5381**	2.83%	1.02%
Task 3 ($k=32$)	DoubanMovie (sparser)	**.6530***	.3631	.6137	.3460	.6310	.3776	**.6598**	**.4087**	1.04%	9.48%
	MovieLens (richer)	**.8243***	.5213	.8111	.5167	.8301	.5280	**.8612**	**.5478**	4.48%	4.72%
Task 3 ($k=64$)	DoubanMovie (sparser)	**.6477***	.3605	.6200	.3544	.6423	.3841	**.6654**	**.4101**	2.73%	9.24%
	MovieLens (richer)	.8200	**.5382***	.8130	.5198	.8324	.5320	**.8668**	**.5516**	5.31%	2.50%
Task 3 ($k=128$)	DoubanMovie (sparser)	**.6521***	.3642	.6222	.3714	.6489	.3792	**.6812**	**.4198**	4.46%	6.90%
	MovieLens (richer)	.8267	**.5401***	.8210	.5311	.8349	.5381	**.8642**	**.5512**	3.88%	2.06%

larger embedding can represent a user/item more accurately. However, in light of the structure of the neural network layers in the **Parameter Setting**, the training time of the GA-DTCDR also increased with k. This is a trade-off. Therefore, in considering both aspects, $k = 64$ is ideal in this thesis's experiments.

Summary 3: Overall, the embedding dimension k is a sensitive parameter for recommendation performance and training time. Both the recommendation performance and the training time of the GA-DTCDR increased with k.

5.6.2.4 Result 4: Top-N Recommendation Performance (for Q4)

To answer **Q4**, the performance of top-N recommendation was compared in terms of HR@N and NDCG@N, where N ranges from 1 to 10. The performance trends of all top-N experiments (for all tasks with different k) are similar. Therefore, due to space limitation, only the top-N recommendation results were reported for all sev-



(a) DoubanBook (HR@N) (b) DoubanBook (NDCG@N)

(c) DoubanMovie (HR@N) (d) DoubanMovie (NDCG@N)

Figure 5.4: The result of Top-N recommendation for Task 1 ($k = 8$)

en baseline models, GA-DTCDR_Average and GA-DTCDR for Task 1 ($k = 8$), as shown in Figure 5.4. In Figure 5.4, in both the DoubanBook (sparser) and DoubanMovie (richer) categories, the performance of the GA-DTCDR is consistently better than those of all seven baselines and GA-DTCDR_Average. For the DoubanBook category, considering all top-N recommendations, this thesis's GA-DTCDR improved the best-performing baselines in different experimental cases by an average of 1.74% for HR@N and 5.83% for NDCG@N. For the DoubanMovie category, the GA-DTCDR improved the best-performing baselines in different experimental cases by an average of 8.13% for HR@N and 7.55% for NDCG@N.

Summary 4: Overall, this thesis's GA-DTCDR outperformed the eight baseline models for top-N recommendation. It was generally found that the larger the N, then the better the improvements the GA-DTCDR achieved.

5.7 Summary

In this chapter, a Graphical and Attention framework for Dual-Target Cross-Domain Recommendation, called GA-DTCDR, has been proposed, which leverages the data diversity and richness of two domains to improve the recommendation performance on dual-target domains simultaneously. First, the heterogeneous graph is constructed to include all relationships, i.e., user-item, user-user, and item-item relationships. Then, the graph embedding technique is employed to generate more representative embedding matrices of users and items. Based on the element-wise attention mechanism, the embeddings of common users are intelligently combined to improve the recommendation accuracies on both domains. Finally, extensive experiments conducted on real-world datasets have demonstrated the superior performance of our GA-DTCDR.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

Although CF has been widely used and has proven to be one of the most promising techniques in many RSs, data sparsity is yet a long-standing problem in these systems. This is because few of the users can provide enough ratings or reviews to many items, which reduces the recommendation accuracy of existing CF-based models. This is e-specially true for new users, who do not provide feedback to any items and exemplify the cold-start problem (a special case of data sparsity). To address the data sparsity problem, CDR was proposed to leverage the auxiliary information from a source domain with richer information to improve the recommendation accuracy in a target domain with sparser information.

However, existing single-target CDR approaches have difficulty in accurately transferring knowledge from the source domain to the target domain. This leads to this thesis's first challenge **CH1**: *'How can an accurate mapping of the latent factors across domains be found for enhancing recommendation accuracy?'*.

Additionally, existing single-target CDR approaches can only improve the recommendation accuracy in the target domain by leveraging the auxiliary information from the source domain. This means that the source domain cannot be further improved by leveraging the useful information from the target domain. However, it is intuitively possible to improve the recommendation accuracy in both domains simultaneously by using dual-target CDR, if the certain types of richer information from both domains can be leveraged effectively.

Achieving dual-target CDR involves facing the remaining three challenges identified in this thesis: **CH2**: *'how can a feasible framework for dual-target CDR be devised?'*; **CH3**: *'how can the data richness and diversity be leveraged to generate more representative single-domain user and item embeddings for improving recommendation accuracy in both domains?'*; and **CH4**: *'how can the user or item embeddings be effectively optimised in each target domain for improving recommendation accuracies in both domains?'*.

To target these four challenges, this thesis has proposed certain solutions:

- In targeting **CH1** in Chapter 3, this thesis has proposed the DCDCSR framework for both CDR and CSR; it is based on MF models and a fully connected DNN, which is applied to more accurately map the latent factors across domains or systems. Additionally, this thesis used the sparsity degrees of individual users and items in the source and target domains or systems to guide the DNN training process, which could effectively utilise more rating data. The superior performances of this thesis's model have been demonstrated by the extensive experiments conducted on three real-world datasets.

- In targeting **CH2** and **CH3** in Chapter 4, the general framework of DTCDR was proposed, which leverages ratings and multi-source content to improve the recommendation performance on dual-target domains simultaneously. Document embedding and rating embedding techniques were optimised to generate the text and rating embeddings of users and items. Based on MTL, a flexible and effective embedding-sharing strategy was adopted to combine and share the embeddings of common users across domains. Finally, extensive experiments that were conducted on real-world datasets have demonstrated the superior performance of this thesis's models.

- To target **CH2**, **CH3** and **CH4**, in Chapter 5, the GA-DTCDR was proposed. This framework utilised the graph-embedding technique to generate more representative user and item embeddings, as well as the element-wise attention mechanism to improve the recommendation accuracies in both domains simultaneously. Extensive experiments were also conducted to demonstrate the superior performance of this thesis's GA-DTCDR.

6.2 Future Work

This thesis has mainly focused on studying CDR and dual-target CDR problems. Three frameworks have been proposed to address the four challenges listed earlier, and extensive experiments have been conducted to validate the performance of this thesis's proposed approaches. However, there are still unresolved issues, so any future work should thus consider the following aspects:

- In Chapter 3, a deep framework for both CDR and CSR was proposed, one that can accurately map the latent factors across domains or systems. It should be a future goal to propose a more effective combination strategy that can generate benchmark latent factors for accurate mapping. For example, attention networks could be a good option to combine the latent factors learned from both domains and generate more reasonable benchmark latent factors. To further improve mapping quality, more linear and non-linear mapping strategies (e.g., deep & wide learning and graph neural networks) could be used. Additionally, it should be promising to devise a hybrid framework which can utilise the auxiliary information from both a source domain and a source system to further improve the recommendation accuracy in a target domain or system — CDR + CSR.
- A dual-target CDR framework was proposed in Chapter 4 that can improve the recommendation performance in both domains simultaneously. In future studies, the approach to multi-target recommendations should be extended — that tends to improve the recommendation performance in multiple domains simultaneously. This multi-target recommendation problem will face new challenges,

e.g., multi-target recommendation systems may heavily rely on overlaps (common users/items) and rich historical data. Therefore, the effect of the proportion of common users between multiple domains and the sparsity of datasets on recommendation performance should be studied.

- A graphical and attention framework for dual-target CDR has also been proposed in Chapter 5, which can employ the graph-embedding strategy and element-wise attention mechanism. In the future, the graph-embedding strategy could be further improved by constructing a more informative graph. For example, social relations or more side information could be used to construct a graph. The attention mechanism could consider devising a self-attention network to generate more representative combined embeddings. Additionally, the approach to multi-target recommendations could be extended, and more comprehensive experiments could be conducted on new datasets to validate the effect of the data sparsity and the scale of common users on performance.

These aspects are very important for single-target CDR and dual-target CDR. Particularly, the first point includes two aspects, i.e., more mapping strategies and CDR + CSR, which aim to utilise auxiliary data more reasonably and to utilise more auxiliary data by more overlaps, i.e., common users + common items, respectively. The second and third points mainly include two aspects, i.e., multi-target recommendation and multi-content recommendation, which aim to utilise more auxiliary data from more domains and to utilise more types of auxiliary data, respectively. The purest motivation behind these aspects is to properly utilise more types of data from more related domains or systems. If these aspects can be done in the future, the data sparsity problem in RSs will be greatly alleviated and even solved. Then, the recommendation accuracy of many recommender systems will be significantly improved.

Appendix A

The Notations in the Thesis

Table A.1: The important notations in Chapter 4 (part 1)

Symbol	Definition
$c_{ij} \in C$	the comment (e.g., the review and the tags) of user u_i on item v_j
$C \in \mathbb{R}^{m \times n}$	the user comments
$D = \{d_1, d_2, ..., d_{m+n}\}$	the content documents of users and items
$ID = \{id_1, ..., id_n\}$	the item details
k	the dimension of embedding
m	the number of users
n	the number of items
P	the optimised embedding of users
Q	the optimised embedding of items
$r_{ij} \in R$	the rating of user u_i on item v_j
$R \in \mathbb{R}^{m \times n}$	the rating matrix

Table A.2: The important notations in Chapter 4 (part 2)

Symbol	**Definition**
$\mathcal{U} = \{u_1, ..., u_m\}$	the set of users
U	the rating embedding of users
UC	the document embedding of users
$UP = \{up_1, ..., up_m\}$	the user profiles
$\mathcal{V} = \{v_1, ..., v_n\}$	the set of items
V	the rating embedding of items
VC	the document embedding of items
$y_{ij} \in Y$	the interaction of user u_i on item v_j
$Y \in \mathbb{R}^{m \times n}$	the user-item interaction matrix
$*^a$ and $*^b$	the notations in domains A and B, e.g., $\mathcal{U}^a$ represents the set of users $\mathcal{U}$ in domain A
$\hat{*}$	the predicted notations, e.g., $\hat{Y}$ represents the predicted user-item interaction matrix

Table A.3: The important notations in Chapter 5

Symbol	**Definition**
$c_{ij} \in C$	the comment (e.g., the review and the tags) of user u_i on item v_j
$C \in \mathbb{R}^{m \times n}$	the user comments
$D = \{d_1, d_2, ..., d_{m+n}\}$	the content documents of users and items
$ID = \{id_1, ..., id_n\}$	the item details
$G = (\{\mathcal{U}, \mathcal{V}\}, E)$	the heterogeneous graph, E is the set of user-user, user-item, and item-item relationships
k	the dimension of embedding matrix
m	the number of users
n	the number of items
$\tilde{U}$	the combined embeddings of common users
$r_{ij} \in R$	the rating of user u_i on item v_j
$R \in \mathbb{R}^{m \times n}$	the rating matrix
$\mathcal{U} = \{u_1, ..., u_m\}$	the set of users
U	the graph embedding matrix of users
UC	the document embedding matrix of users
$UP = \{up_1, ..., up_m\}$	the user profiles
$\mathcal{V} = \{v_1, ..., v_n\}$	the set of items
V	the graph embedding matrix of items
VC	the document embedding matrix of items
$y_{ij} \in Y$	the interaction of user u_i on item v_j
$Y \in \mathbb{R}^{m \times n}$	the user-item interaction matrix
$*^a$ and $*^b$	the notations for domains a and b, e.g., m^a represents the number of users in domain a
$\hat{*}$	the predicted notations, e.g., $\hat{y}_{ij}$ represents the predicted interaction of u_i on item v_j

Appendix B

The Acronyms in the Thesis

Table B.1: The Acronyms in All the Chapters

Sections	**Explanations**	**Acronyms**
Chapter 1&2&4&5	Single-domain recommendtion	SDR
Chapter 1&2&3&4&5&6	Cross-domain recommendtion	CDR
Chapter 1&2&4&5	Multi-domain recommendation	MDR
Chapter 1&2&4&5	Dual-target cross-domain recommendtion	DTCDR
Chapter 1&2&3&4&5	Collaborative filtering	CF
Chapter 1&2&3&4&5	Matrix factorisation	MF
Chapter 1&2&3&4&5	Transfer learning	TL
Chapter 1&2&3&4&5	Multilayer perceptron	MLP
Chapter 1&2&4&5	Multi-task learning	MTL
Chapter 1&2&5	Graph embedding	GE
Chapter 3	Mean absolute error	MAE
Chapter 3	Root mean square error	RMSE
Chapter 4&5	Hit ratio	HR
Chapter 4&5	Normalised discounted cumulative gain	NDCG

Bibliography

[1] F. Abel, Q. Gao, G.-J. Houben, and K. Tao. Analyzing user modeling on twitter for personalized news recommendations. In *international conference on user modeling, adaptation, and personalization*, pages 1–12. Springer, 2011.

[2] A. Agarwal, S. Gerber, and H. Daume. Learning multiple tasks using manifold regularization. In *Advances in neural information processing systems*, pages 46–54, 2010.

[3] D. Agarwal, B.-C. Chen, and B. Long. Localized factor models for multi-context recommendation. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 609–617, 2011.

[4] H. J. Ahn. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. *Information Sciences*, 178(1):37–51, 2008.

[5] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[6] Y. Bao, H. Fang, and J. Zhang. Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In *Twenty-Eighth AAAI conference on artificial intelligence*, 2014.

[7] R. Bell, Y. Koren, and C. Volinsky. Modeling relationships at multiple scales to improve accuracy of large recommender systems. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 95–104, 2007.

[8] S. Berkovsky, T. Kuflik, and F. Ricci. Cross-domain mediation in collaborative filtering. In *International Conference on User Modeling*, pages 355–359. Springer, 2007.

[9] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

[10] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96, 2005.

[11] R. Burke. Hybrid recommender systems: Survey and experiments. *UMUAI*, (4):331–370, 2002.

[12] S. Cao, W. Lu, and Q. Xu. Deep neural networks for learning graph representations. In *Thirtieth AAAI conference on artificial intelligence*, pages 1145–1152, 2016.

[13] R. Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[14] O. Chapelle and S. S. Keerthi. Efficient algorithms for ranking with svms. *Information retrieval*, 13(3):201–215, 2010.

[15] J. Chen, J. Liu, and J. Ye. Learning incoherent sparse and low-rank patterns from multiple tasks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(4):1–31, 2012.

[16] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 335–344, 2017.

[17] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image

captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5659–5667, 2017.

[18] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, et al. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, pages 7–10, 2016.

[19] R. Chung, D. Sundaram, and A. Srinivasan. Integrated personal recommender systems. In *ICEC*, pages 65–74, 2007.

[20] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

[21] A. M. Elkahky, Y. Song, and X. He. 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*, pages 278–288, 2015.

[22] A. Farseev, I. Samborskii, A. Filchenkov, and T.-S. Chua. Cross-domain recommendation via clustering on multi-layer graphs. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 195–204, 2017.

[23] I. Fernández-Tobías and I. Cantador. Exploiting social tags in matrix factorization models for cross-domain collaborative filtering. In *CBRecSys@ RecSys*, pages 34–41, 2014.

[24] I. Fernández-Tobías, I. Cantador, M. Kaminskas, and F. Ricci. A generic semantic-based framework for cross-domain recommendation. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems*, pages 25–32, 2011.

[25] I. Fernández-Tobías, I. Cantador, M. Kaminskas, and F. Ricci. Cross-domain recommender systems: A survey of the state of the art. In *Spanish Conference on Information Retrieval*, page 24, 2012.

[26] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of machine learning research*, 4(Nov):933–969, 2003.

[27] 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 *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 94–101, 2019.

[28] C. Gao, X. Chen, F. Feng, K. Zhao, X. He, Y. Li, and D. Jin. Cross-domain recommendation without sharing user-relevant data. In *The World Wide Web Conference*, pages 491–502, 2019.

[29] S. Gao, H. Luo, D. Chen, S. Li, P. Gallinari, and J. Guo. Cross-domain recommendation via cluster-level latent factor model. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 161–176. Springer, 2013.

[30] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AIStats*, pages 249–256, 2010.

[31] G. H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. In *Linear Algebra*, pages 134–151. Springer, 1971.

[32] P. Gong, J. Ye, and C. Zhang. Robust multi-task feature learning. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 895–903, 2012.

[33] A. Grover and J. Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

[34] L. Han and Y. Zhang. Multi-stage multi-task learning with reduced rank. In *Thirtieth AAAI Conference on Artificial Intelligence*, pages 1638–1644, 2016.

[35] F. M. Harper and J. A. Konstan. The movielens datasets: History and context. *TIIS*, 5(4):19, 2016.

[36] J. He, R. Liu, F. Zhuang, F. Lin, C. Niu, and Q. He. A general cross-domain recommendation framework via bayesian neural network. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 1001–1006. IEEE, 2018.

[37] M. He, J. Zhang, P. Yang, and K. Yao. Robust transfer learning for cross-domain collaborative filtering using multiple rating patterns approximation. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 225–233, 2018.

[38] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182, 2017.

[39] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu. Leveraging meta-path based context for top-n recommendation with a neural co-attention model. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1531–1540, 2018.

[40] G. Hu, Y. Zhang, and Q. Yang. Conet: Collaborative cross networks for cross-domain recommendation. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 667–676, 2018.

[41] G. Hu, Y. Zhang, and Q. Yang. Transfer meets hybrid: A synthetic approach for cross-domain collaborative filtering with text. In *The World Wide Web Conference*, pages 2822–2829, 2019.

[42] L. Hu, J. Cao, G. Xu, L. Cao, Z. Gu, and C. Zhu. Personalized recommendation via cross-domain triadic factorization. In *WWW*, pages 595–606, 2013.

[43] L. Huang, Z.-L. Zhao, C.-D. Wang, D. Huang, and H.-Y. Chao. Lscd: Low-rank and sparse cross-domain recommendation. *Neurocomputing*, 366:86–96, 2019.

[44] A. Jalali, S. Sanghavi, C. Ruan, and P. K. Ravikumar. A dirty model for multi-task learning. In *Advances in neural information processing systems*, pages 964–972, 2010.

[45] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich. *Recommender systems: an introduction*. 2010.

[46] S. Jaradat. Deep cross-domain fashion recommendation. In *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pages 407–410, 2017.

[47] M. Kaminskas and F. Ricci. Location-adapted music recommendation using tags. In *International conference on user modeling, adaptation, and personalization*, pages 183–194. Springer, 2011.

[48] H. Kanagawa, H. Kobayashi, N. Shimizu, Y. Tagami, and T. Suzuki. Cross-domain recommendation via deep domain adaptation. In *European Conference on Information Retrieval*, pages 20–29. Springer, 2019.

[49] S. Kang, J. Hwang, D. Lee, and H. Yu. Semi-supervised learning for cross-domain recommendation to cold-start users. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1563–1572, 2019.

[50] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[51] T. N. Kipf and M. Welling. Variational graph auto-encoders. *arXiv preprint arXiv:1611.07308*, 2016.

[52] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434, 2008.

[53] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[54] J. B. Kruskal and M. Wish. *Multidimensional scaling*, volume 11. 1978.

[55] A. Kumar, N. Kumar, M. Hussain, S. Chaudhury, and S. Agarwal. Semantic clustering-based cross-domain recommendation. In *2014 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, pages 137–141. IEEE, 2014.

[56] Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *ICML*, pages 1188–1196, 2014.

[57] C. W.-k. Leung, S. C.-f. Chan, and F.-l. Chung. Applying cross-level association rule mining to cold-start recommendations. In *2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops*, pages 133–136. IEEE, 2007.

[58] B. Li, Q. Yang, and X. Xue. Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. In *IJCAI*, pages 2052–2057, 2009.

[59] B. Li, Q. Yang, and X. Xue. Transfer learning for collaborative filtering via a rating-matrix generative model. In *Proceedings of the 26th annual international conference on machine learning*, pages 617–624, 2009.

[60] B. Li, X. Zhu, R. Li, C. Zhang, X. Xue, and X. Wu. Cross-domain collaborative filtering over time. In *Twenty-Second International Joint Conference on Artificial Intelligence*, pages 2293–2298, 2011.

[61] C.-Y. Li and S.-D. Lin. Matching users and items across domains to improve the recommendation quality. In *SIGKDD*, pages 801–810, 2014.

[62] L. Li, Q. Do, and W. Liu. Cross-domain recommendation via coupled factorization machines. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9965–9966, 2019.

[63] P. Li and A. Tuzhilin. Ddtcdr: Deep dual transfer cross domain recommendation. *arXiv preprint arXiv:1910.05189*, 2019.

[64] W. Lian, R. Henao, V. Rao, J. Lucas, and L. Carin. A multitask point process predictive model. In *International Conference on Machine Learning*, pages 2030–2038, 2015.

[65] G. Ling, M. R. Lyu, and I. King. Ratings meet reviews, a combined approach to recommend. In *RecSys*, pages 105–112, 2014.

[66] B. Liu, Y. Wei, Y. Zhang, Z. Yan, and Q. Yang. Transferable contextual bandit for cross-domain recommendation. In *Thirty-Second AAAI Conference on Artificial Intelligence*, pages 3619–3626, 2018.

[67] J. Liu, P. Zhao, F. Zhuang, Y. Liu, V. S. Sheng, J. Xu, X. Zhou, and H. Xiong. Exploiting aesthetic preference in deep cross networks for cross-domain recommendation. In *Proceedings of The Web Conference 2020*, pages 2768–2774, 2020.

[68] B. Loni, Y. Shi, M. Larson, and A. Hanjalic. Cross-domain collaborative filtering with factorization machines. In *European conference on information retrieval*, pages 656–661. Springer, 2014.

[69] Y. Lu, R. Dong, and B. Smyth. Why i like it: multi-task learning for recommendation and explanation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 4–12, 2018.

[70] M. Ma, P. Ren, Y. Lin, Z. Chen, J. Ma, and M. d. Rijke. π-net: A parallel information-sharing network for shared-account cross-domain sequential recommendations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 685–694, 2019.

[71] T. Man, H. Shen, X. Jin, and X. Cheng. Cross-domain recommendation: An embedding and mapping approach. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*, pages 2464–2470, 2017.

[72] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The Stanford CoreNLP toolkit. In *ACL System Demonstrations*, pages 55–60, 2014.

[73] J. Manotumruksa, D. Rafailidis, C. Macdonald, and I. Ounis. On cross-domain transfer in venue recommendation. In *European Conference on Information Retrieval*, pages 443–456. Springer, 2019.

[74] J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *RecSys*, pages 165–172, 2013.

[75] L. Mihalkova, T. Huynh, and R. J. Mooney. Mapping and revising markov logic networks for transfer learning. In *Twenty-Second AAAI Conference on Artificial Intelligence*, pages 608–614, 2007.

[76] L. Mihalkova and R. J. Mooney. Transfer learning by mapping with minimal target data. In *Proceedings of the AAAI-08 workshop on transfer learning for complex tasks*, pages 31–36, 2008.

[77] T. Mikolov, Q. V. Le, and I. Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*, 2013.

[78] A. Mnih and R. R. Salakhutdinov. Probabilistic matrix factorization. In *Advances in neural information processing systems*, pages 1257–1264, 2008.

[79] R. J. Mooney and L. Roy. Content-based book recommending using learning for text categorization. In *Proceedings of the fifth ACM conference on Digital libraries*, pages 195–204, 2000.

[80] O. Moreno, B. Shapira, L. Rokach, and G. Shani. Talmud: transfer learning for multiple domains. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 425–434, 2012.

[81] S. Pan, R. Hu, G. Long, J. Jiang, L. Yao, and C. Zhang. Adversarially regularized graph autoencoder for graph embedding. pages 2609–2615, 2018.

[82] S. J. Pan and Q. Yang. A survey on transfer learning. *TKDE*, 22(10):1345–1359, 2009.

[83] W. Pan, N. N. Liu, E. W. Xiang, and Q. Yang. Transfer learning to predict missing ratings via heterogeneous user feedbacks. In *Twenty-Second International Joint Conference on Artificial Intelligence*, number 3, pages 2318–2323, 2011.

[84] W. Pan, E. W. Xiang, N. N. Liu, and Q. Yang. Transfer learning in collaborative filtering for sparsity reduction. In *Twenty-fourth AAAI conference on artificial intelligence*, 2010.

[85] W. Pan and Q. Yang. Transfer learning in heterogeneous collaborative filtering domains. *Artificial intelligence*, 197:39–55, 2013.

[86] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.

[87] I. Porteous, E. Bart, and M. Welling. Multi-hdp: A non parametric bayesian model for tensor factorization. In *Twenty-Third AAAI Conference on Artificial Intelligence*, pages 1487–1490, 2008.

[88] D. Rafailidis and F. Crestani. Top-n recommendation via joint cross-domain user clustering and similarity learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 426–441. Springer, 2016.

[89] D. Rafailidis and F. Crestani. A collaborative ranking model for cross-domain recommendations. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 2263–2266, 2017.

[90] S. Ren, S. Gao, J. Liao, and J. Guo. Improving cross-domain recommendation through probabilistic cluster-level latent factor model. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[91] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, pages 452–461, 2009.

[92] F. Ricci, L. Rokach, and B. Shapira. *Recommender Systems Handbook*. 2nd edition, 2015.

[93] M. Riedmiller and H. Braun. A direct adaptive method for faster backpropagation learning: The rprop algorithm. In *ICNN*, pages 586–591, 1993.

[94] S. Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

[95] S. Sahebi and T. Walker. Content-based cross-domain recommendations using segmented models. In *CBRecSys@ RecSys*, pages 57–64, 2014.

[96] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *WWW*, pages 285–295, 2001.

[97] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 253–260, 2002.

[98] J. Shang, M. Sun, and K. Collins-Thompson. Demographic inference via knowledge transfer in cross-domain recommender systems. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 1218–1223. IEEE, 2018.

[99] B. Shapira, L. Rokach, and S. Freilikhman. Facebook single and cross domain data for recommendation systems. *UMUAI*, (2-3):211–247, 2013.

[100] X. Shi, Z. Guo, Z. Lai, Y. Yang, Z. Bao, and D. Zhang. A framework of joint graph embedding and sparse regression for dimensionality reduction. *IEEE Transactions on Image Processing*, 24(4):1341–1355, 2015.

[101] Y. Shi, M. Larson, and A. Hanjalic. Tags as bridges between domains: Improving recommendation with tag-induced cross-domain collaborative filtering. In *International Conference on User Modeling, Adaptation, and Personalization*, pages 305–316. Springer, 2011.

[102] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 650–658, 2008.

[103] S. Sopchoke, K.-i. Fukui, and M. Numao. Explainable cross-domain recommendations through relational learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[104] N. Srebro, J. Rennie, and T. S. Jaakkola. Maximum-margin matrix factorization. In *Advances in neural information processing systems*, pages 1329–1336, 2005.

[105] B. W. SUTER. The multilayer perceptron as an approximation to a bayes optimal discriminant function. *IEEE Transactions on Neural Networks*, 1(4):291, 1990.

[106] M. Szomszor, H. Alani, I. Cantador, K. OHara, and N. Shadbolt. Semantic modelling of user interests based on cross-folksonomy analysis. In *International Semantic Web Conference*, pages 632–648. Springer, 2008.

[107] G. Takács, I. Pilászy, B. Németh, and D. Tikk. Scalable collaborative filtering approaches for large recommender systems. *Journal of machine learning research*, 10(Mar):623–656, 2009.

[108] S. Tan, J. Bu, X. Qin, C. Chen, and D. Cai. Cross domain recommendation based on multi-type media fusion. *Neurocomputing*, 127:124–134, 2014.

[109] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077, 2015.

[110] J. Tang, S. Wu, J. Sun, and H. Su. Cross-domain collaboration recommendation. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1285–1293, 2012.

[111] Y. Tay, A. T. Luu, and S. C. Hui. Multi-pointer co-attention networks for recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2309–2318, 2018.

[112] S. Thrun and J. O'Sullivan. Discovering structure in multiple learning tasks: The tc algorithm. In *13th International Conference on Machine Learning*, pages 489–497, 1996.

[113] M. K. Titsias and M. Lázaro-Gredilla. Spike and slab variational inference for multi-task and multiple kernel learning. In *Advances in neural information processing systems*, pages 2339–2347, 2011.

[114] K. Tu, P. Cui, X. Wang, P. S. Yu, and W. Zhu. Deep recursive network embedding with regular equivalence. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2357–2366, 2018.

[115] B. Wang, M. Ester, Y. Liao, J. Bu, Y. Zhu, Z. Guan, and D. Cai. The million domain challenge: Broadcast email prioritization by cross-domain recommendation. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1895–1904, 2016.

[116] C. Wang and D. M. Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 448–456, 2011.

[117] D. Wang, P. Cui, and W. Zhu. Structural deep network embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1225–1234, 2016.

[118] X. Wang, X. He, Y. Cao, M. Liu, and T.-S. Chua. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 950–958, 2019.

[119] Y. Wang, C. Feng, C. Guo, Y. Chu, and J.-N. Hwang. Solving the sparsity problem in recommendations via cross-domain item embedding based on co-clustering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 717–725, 2019.

[120] P. Winoto and T. Tang. If you like the devil wears prada the book, will you also enjoy the devil wears prada the movie? a study of cross-domain recommendations. *New Generation Computing*, 26(3):209–225, 2008.

[121] S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

[122] X. Xin, Z. Liu, C.-Y. Lin, H. Huang, X. Wei, and P. Guo. Cross-domain collaborative filtering with review text. In *IJCAI*, pages 1827–1834, 2015.

[123] H.-J. Xue, X. Dai, J. Zhang, S. Huang, and J. Chen. Deep matrix factorization models for recommender systems. In *Proceedings of the 26th international joint conference on artificial intelligence*, pages 3203–3209, 2017.

[124] Y. Xue, D. Dunson, and L. Carin. The matrix stick-breaking process for flexible multi-task learning. In *Proceedings of the 24th international conference on Machine learning*, pages 1063–1070, 2007.

[125] S. Yan, D. Xu, B. Zhang, and H.-J. Zhang. Graph embedding: A general framework for dimensionality reduction. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 830–837. IEEE, 2005.

[126] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo. Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4651–4659, 2016.

[127] W. Yu, C. Zheng, W. Cheng, C. C. Aggarwal, D. Song, B. Zong, H. Chen, and W. Wang. Learning deep network representations with adversarially regularized autoencoders. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2663–2671, 2018.

[128] F. Yuan, L. Yao, and B. Benatallah. Darec: Deep domain adaptation for cross-domain recommendation via transferring rating patterns. *arXiv preprint arXiv:1905.10760*, 2019.

[129] J. Zhang, Z. Ghahramani, and Y. Yang. Learning multiple related tasks using latent independent component analysis. In *Advances in neural information processing systems*, pages 1585–1592, 2006.

[130] Q. Zhang, P. Hao, J. Lu, and G. Zhang. Cross-domain recommendation with semantic correlation in tagging systems. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[131] Q. Zhang, J. Lu, D. Wu, and G. Zhang. A cross-domain recommender system with kernel-induced knowledge transfer for overlapping entities. *IEEE transactions on neural networks and learning systems*, 30(7):1998–2012, 2018.

[132] Q. Zhang, D. Wu, J. Lu, F. Liu, and G. Zhang. A cross-domain recommender system with consistent information transfer. *Decision Support Systems*, 104:49–63, 2017.

[133] Q. Zhang, D. Wu, J. Lu, and G. Zhang. Cross-domain recommendation with probabilistic knowledge transfer. In *International Conference on Neural Information Processing*, pages 208–219. Springer, 2018.

[134] W. Zhang, R. Li, T. Zeng, Q. Sun, S. Kumar, J. Ye, and S. Ji. Deep model based transfer and multi-task learning for biological image analysis. *IEEE transactions on Big Data*, 2016.

[135] Y. Zhang, B. Cao, and D.-Y. Yeung. Multi-domain collaborative filtering. *arXiv preprint arXiv:1203.3535*, 2012.

[136] Y. Zhang and Q. Yang. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*, 2017.

[137] Z. Zhang, X. Jin, L. Li, G. Ding, and Q. Yang. Multi-domain active learning for recommendation. In *Thirtieth AAAI Conference on Artificial Intelligence*, pages 2358–2364, 2016.

[138] L. Zhao, S. J. Pan, E. W. Xiang, E. Zhong, Z. Lu, and Q. Yang. Active transfer learning for cross-system recommendation. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[139] L. Zhao, S. J. Pan, and Q. Yang. A unified framework of active transfer learning for cross-system recommendation. *Artificial Intelligence*, 245:38–55, 2017.

[140] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, and M. Sun. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*, 2018.

[141] F. Zhu, C. Chen, Y. Wang, G. Liu, and X. Zheng. Dtcdr: A framework for dual-target cross-domain recommendation. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1533–1542. ACM, 2019.

[142] F. Zhu, Y. Wang, C. Chen, G. Liu, M. A. Orgun, and J. Wu. A deep framework for cross-domain and cross-system recommendations. In *IJCAI International Joint Conference on Artificial Intelligence*, pages 3711–3717, 2018.

[143] F. Zhu, Y. Wang, C. Chen, G. Liu, and X. Zheng. Graphical and attentional framework for dual-target cross-domain recommendation. 2020.