修 士 論 文

Distributional User Preference Modeling for Cross-Domain Recommendation under Non-Overlapping Settings

非重複設定下におけるクロスドメイン推薦のための分布的
ユーザ嗜好モデリング

指導教員 　　鈴村 豊太郎 教授

東京大学大学院 情報理工学系研究科
電子情報学専攻

氏 名 　48-246453 肖　子吟

提 出 日 　　2026 年 1 月 22 日

Abstract

Recommendation systems play a crucial role in helping users discover relevant items across a wide range of applications, including e-commerce, streaming services, and social media platforms. With the rapid expansion of online platforms and available content, developing effective recommendation systems has become increasingly important.

Despite their success, traditional recommendation systems often suffer from data sparsity and cold-start problems, which significantly limit their performance. Cross-domain recommendation has therefore emerged as a promising paradigm that leverages information from multiple domains to enhance recommendation accuracy. However, most existing cross-domain recommendation methods rely on the assumption of overlapping users or items across domains, an assumption that is frequently violated in real-world scenarios. Moreover, user preferences are typically represented as point estimates, which are insufficient for capturing the uncertainty and multi-faceted nature of user behavior.

In this thesis, we propose a novel cross-domain recommendation framework that models user preferences as Gaussian Mixture Models (GMMs) and integrates optimal transport to enable knowledge transfer under non-overlapping settings. By representing user preferences as probability distributions, the proposed method effectively captures the diversity and uncertainty inherent in user interests. Furthermore, optimal transport provides a principled mechanism for aligning preference distributions across domains, allowing effective knowledge transfer even in the absence of overlapping users or items.

Extensive experiments on multiple Amazon datasets demonstrate that the proposed approach consistently outperforms state-of-the-art cross-domain recommendation methods as well as strong single-domain baselines. Additional ablation studies further verify the importance of distributional user preference modeling and the effectiveness of incorporating source-domain information to enhance target-domain recommendation performance.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1   Background

Information overload has become a critical challenge for users on modern online platforms, including e-commerce websites, streaming services, and social media [1, 2, 3]. Recommender systems play a crucial role in alleviating this problem by providing personalized guidance, selecting items with the highest predicted utility from a large pool of candidates [4, 5, 6]. By analyzing historical user–item interactions, recommender systems help users discover relevant content, products, or services, thereby improving user satisfaction and engagement. Beyond enhancing user experience, recommender systems also generate substantial business value by increasing sales, customer retention, and overall platform activity [7]. As the scale and diversity of online content continue to grow, the importance of effective recommender systems becomes increasingly pronounced.

   Despite their success, recommender systems face several fundamental challenges that limit their performance. One major challenge is data sparsity, where users interact with only a small fraction of available items, making it difficult to accurately infer user preferences. This issue is particularly severe in emerging or niche platforms with limited interaction data. Another critical challenge is the cold-start problem, which arises when new users or items enter the system and insufficient historical data is available to support reliable recommendations. Both data sparsity and cold-start problems stem from the limited availability of user–item interaction data and can significantly degrade recommendation quality.

   To address these challenges, cross-domain recommendation has emerged as a promising paradigm that leverages information from a related source domain to improve recommendation performance in a target domain. By transferring knowledge across domains, cross-domain recommendation is able to alleviate data sparsity and cold-start issues, particularly in data-scarce settings. However, most existing cross-domain recommendation methods rely on the assumption that users or items overlap between the source and target domains, and such overlaps are used as bridges for knowledge transfer during the training stage. In many real-world scenarios, however, this assumption does not hold, as domains may be entirely disjoint due to privacy constraints, system isolation, or independent platform design. Moreover, traditional recommender systems typically represent user preferences as

point estimates in a latent space, which are often insufficient to capture the uncertainty and multi-faceted nature of real user behavior.

Motivated by these limitations, this thesis proposes a novel cross-domain recommendation framework that models user preferences as Gaussian Mixture Models (GMMs) and integrates optimal transport techniques to enable knowledge transfer under non-overlapping settings. By representing user preferences as probability distributions rather than point estimates, the proposed approach captures both the diversity and uncertainty of user interests. Furthermore, optimal transport provides a principled mechanism for aligning user preference distributions across domains, facilitating effective knowledge transfer even when no overlapping users or items are available during training.

The proposed framework consists of two key components: (1) modeling user preferences as Gaussian mixture distributions to capture the complex structure of user behavior within each domain, and (2) employing optimal transport to align these distributions between the source and target domains. Together, these components enable robust cross-domain recommendation and improve recommendation accuracy in the target domain under strictly non-overlapping training settings.

1.2   Contributions

The main contributions of this thesis are as follows:

- We introduce a novel framework for cross-domain recommendation under non-overlapping settings by modeling

- We propose representing user preferences as Gaussian Mixture Models (GMMs) to capture the uncertainty and diversity of user behavior more effectively.

- We integrate optimal transport techniques to align user preference distributions across domains, enabling knowledge transfer

- We conduct extensive experiments on Amazon datasets across multiple domains to evaluate the performance of our proposed method. The results demonstrate that our approach significantly outperforms existing cross-domain recommendation techniques and traditional single-domain methods in terms of recommendation accuracy.

- We perform ablation studies to assess the effectiveness of representing user preferences as distributions and the impact of boosting target domain recommendation performance using source domain information.

- We provide insights into the benefits of modeling user preferences as distributions and leveraging optimal transport for cross-domain knowledge transfer, highlighting the potential of our approach for addressing challenges in recommender systems.

1.3   Thesis Organization

The remainder of this thesis is organized as follows:

- In Chapter 2, we provide an overview of the fundamental concepts and techniques relevant to this research, including recommender systems, cross-domain recommendation, Gaussian Mixture Models, and optimal transport.

- In Chapter 3, we review related work on recommender systems, cross-domain recommendation, and optimal transport techniques.

- In Chapter 4, we present our proposed framework for cross-domain recommendation under non-overlapping settings, detailing the modeling of user preferences as GMMs and the integration of optimal transport for knowledge transfer.

- In Chapter 5, we describe the experimental setup, datasets, and evaluation metrics used to assess the performance of our proposed method.

- In Chapter 6, we present and discuss the experimental results, including comparisons with existing methods and ablation studies.

- Finally, in Chapter 7, we summarize the main findings of this thesis and discuss potential directions for future research.

Chapter 2

Preliminaries

This chapter introduces the fundamental concepts and mathematical formulations that underpin our proposed method, including recommender systems, cross-domain recommendation, Gaussian mixture models, and optimal transport. These preliminaries provide the necessary background for understanding the distributional preference modeling and cross-domain alignment techniques presented in later chapters.

2.1   Recommender Systems

A recommender system aims to predict users' preferences for items and to provide personalized item rankings. Formally, let U denote the set of users and I denote the set of items. A recommender system can be modeled as a function

$$R : U \times I \to S,$$

where S is the space of possible scores or ratings. Given a user $u \in U$ and an item $i \in I$, the system predicts a score $\hat{s}_{ui} = R(u, i)$ that reflects the user's preference for the item.

Based on the predicted scores, the recommender system produces a ranked list of items for each user. The primary objective is to rank items such that those with higher predicted utility are recommended earlier, thereby maximizing user satisfaction and engagement.

2.2   Cross-Domain Recommendation

Cross-domain recommendation extends the standard recommendation setting by leveraging information from one or more source domains to improve recommendation performance in a target domain. Let $D_s = (U_s, I_s)$ denote a source domain with user set U_s and item set I_s, and let $D_t = (U_t, I_t)$ denote a target domain with user set U_t and item set I_t. The goal of cross-domain recommendation is to learn a target-domain recommendation function

$$R_t : U_t \times I_t \to S$$

by exploiting auxiliary information available in the source domain D_s.

The transferable information across domains may include user–item interaction data, user or item attributes, textual reviews, or latent representations learned from the source domain. A central challenge in cross-domain recommendation lies in effectively transferring useful knowledge while accounting for domain heterogeneity, data sparsity, and cold-start issues. In particular, many existing methods rely on overlapping users or items to establish correspondences between domains, an assumption that does not always hold in real-world applications.

2.3   Gaussian Mixture Models

Gaussian Mixture Models (GMMs) are probabilistic models that represent complex distributions as weighted combinations of multiple Gaussian components. A GMM with K components defines a probability density function of the form

$$p(x) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(x \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$

where π_k denotes the mixture weight of the k-th component, satisfying $\pi_k \geq 0$ and $\sum_{k=1}^{K} \pi_k = 1$. Each component $\mathcal{N}(x \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ is a Gaussian distribution parameterized by a mean vector $\boldsymbol{\mu}_k$ and a covariance matrix $\boldsymbol{\Sigma}_k$.

GMMs are capable of modeling multi-modal and heterogeneous data distributions, making them well suited for representing complex phenomena such as diverse and multi-faceted user preferences. In this thesis, GMMs are used to model user preferences as probability distributions rather than single point estimates.

2.4   Optimal Transport and Wasserstein Distance

Optimal transport (OT) is a mathematical framework for comparing probability distributions by explicitly accounting for the cost of transporting probability mass between them. Given two probability measures μ and ν defined on spaces X and Y, respectively, and a ground cost function $c(x, y)$ that measures the cost of moving mass from $x \in X$ to $y \in Y$, the Kantorovich formulation of optimal transport is defined as

$$\inf_{\gamma \in \Pi(\mu,\nu)} \int_{X \times Y} c(x, y) \, d\gamma(x, y),$$

where $\Pi(\mu, \nu)$ denotes the set of all joint probability measures whose marginals are μ and ν.

A widely used metric derived from optimal transport is the Wasserstein distance. For $p \geq 1$, the p-Wasserstein distance between μ and ν is defined as

$$W_p(\mu, \nu) = \left(\inf_{\gamma \in \Pi(\mu,\nu)} \int_{X \times Y} c(x, y)^p \, d\gamma(x, y) \right)^{1/p}.$$

Unlike conventional pointwise distances such as the Euclidean distance, the Wasserstein distance captures the underlying geometric structure of probability distributions and provides meaningful comparisons even when the distributions have limited or no overlapping support.

In this thesis, we adopt the Wasserstein distance as a fundamental tool for measuring discrepancies between distributional user preference representations across domains. By operating at the distribution level, optimal transport enables principled cross-domain alignment that is particularly suitable for non-overlapping recommendation settings.

Chapter 3

Related Work

3.1   Recommender Systems

This section reviews the existing literature on recommender systems, categorizing the related work into several key areas based on the techniques and approaches employed. Each subsection delves into specific methodologies and advancements within the field.

3.1.1   Traditional Recommender Systems

In the early stages of recommender systems, traditional techniques such as collaborative filtering, content-based filtering, and hybrid methods were predominantly used. Collaborative filtering (CF) is one of the foundational approaches in recommender systems, operating on the principle that users with similar preferences will like similar items, or that items liked by similar users will be preferred by a given user [8, 9].

CF can be divded into memory-based and model-based methods based on how recommendations are generated. Memory-based methods utilize user-item interaction data directly to compute similarities between users or items, while model-based methods employ machine learning algorithms to learn latent factors from the interaction data.

3.1.1.1   Memory-based CF

By user or item similarities, memory-based CF can be further categorized into user-based and item-based approaches. User-based CF recommends items to a user based on the preferences of similar users, while item-based CF suggests items similar to those the user has previously liked.

GroupLens [10] is one of the earliest and most influential memory-based CF systems, which introduced user-based collaborative filtering using Pearson correlation to compute user similarities. It demonstrated the effectiveness of CF in providing personalized recommendations and laid the groundwork for subsequent research in the field.

Sarwar et al. [11] points out that when the number of users and items is very large, user-based CF can be computationally expensive and the sparsity of the user-item interaction matrix can lead

to poor recommendation quality. Noticing that users' preferences change quickly over time but items' characteristics are relatively stable, they proposed an item-based CF approach that computes item similarities based on user interactions. The merit of this method is that item similarities can be precomputed and stored, allowing for efficient recommendation generation. And the number of items is usually much smaller than the number of users, which helps alleviate the data sparsity issue. Also, one important attribution of this paper is that it proposed a new similarity measure called adjusted cosine similarity, which accounts for individual user rating biases when computing item similarities. This method has since become a standard technique in item-based CF and has been widely adopted in various recommender systems.

3.1.1.2   Model-based CF

Model-based CF methods utilize machine learning techniques to learn latent representations of users and items from interaction data, by fitting parametric models such as matrix factorization, probabilistic latent factor models, or neural networks. These models capture underlying preference patterns in a low-dimensional latent space, enabling generalization to unseen user – item pairs and alleviating data sparsity.

Breese et al. [12] conducted a comprehensive analysis of both memory-based and model-based CF methods. This paper is the first to systematically distinguish between memory-based and model-based CF approaches, providing a detailed comparison of their strengths and weaknesses. The model-based methods mentioned in this paper include Bayesian Clustering and Bayesian Networks. Bayesian Clustering groups users into clusters based on their preferences, while Bayesian Networks model the probabilistic relationships between users and items.

Ungar and Foster [13] pointed out that traditional clustering-based collaborative filtering methods suffer from instability and poor generalization when interaction data are highly sparse, as approaches based on KNN or simple K-means clustering rely heavily on local similarity patterns. To address this issue, they proposed Gibbs clustering for collaborative filtering, a probabilistic co-clustering approach that jointly clusters users and items into latent classes and models their interactions through class-level link probabilities. By explicitly formulating a generative model and employing Gibbs sampling for inference, their method enforces global consistency in user and item assignments and provides a principled alternative to heuristic clustering.

Koren et al. [14] systematically review matrix factorization techniques for recommender systems, demonstrating that latent factor models with bias, implicit feedback, and temporal dynamics achieve consistently superior accuracy and scalability over neighborhood-based methods, and establishing matrix factorization as a dominant model-based collaborative filtering paradigm.

Salakhutdinov and Mnih introduce Probabilistic Matrix Factorization as a scalable latent factor model for large, sparse recommender systems, and further extend it to a fully Bayesian framework using MCMC, significantly improving robustness and generalization—especially for infrequent users—while establishing PMF as a foundational model-based collaborative filtering paradigm [15, 16].

While most model-based collaborative filtering methods represent user preferences as point embeddings in a latent space, several approaches instead characterize user preferences in a probabilistic manner. Since our proposed method also adopts a distributional representation of user preferences, we briefly review related works along this line.

Hofmann [17] proposed probabilistic latent semantic analysis (PLSA) for collaborative filtering, formulating user – item interactions as a latent class mixture model. In PLSA, each user is associated with a probability distribution over latent topics (or communities), and each interaction is generated by first sampling a latent topic and then drawing an item conditioned on that topic. As a result, user preferences are represented as distributions over latent semantic factors, allowing different interactions of the same user to be explained by different latent causes, rather than being tied to a single latent representation.

Marlin [18] proposed the User Rating Profile (URP) model, a probabilistic latent variable approach for rating-based collaborative filtering that explicitly models uncertainty in user preferences. URP represents each user as a mixture over latent user attitudes, where the mixture proportions are drawn from a Dirichlet distribution. For each item, a latent attitude is sampled and the corresponding rating is generated according to an attitude-specific rating distribution. By modeling users as distributions over latent preference patterns rather than fixed point representations, URP enables different items rated by the same user to be explained by different latent factors and allows direct inference of rating distributions for unseen items.

Blei, Ng, and Jordan [19] proposed Latent Dirichlet Allocation (LDA), a hierarchical generative probabilistic model that represents each document as a mixture over latent topics, where the topic proportions are drawn from a Dirichlet prior. By introducing a document-level latent variable, LDA provides a fully generative framework that enables principled inference for previously unseen data. Although originally developed for text modeling, LDA has been extended to collaborative filtering by treating users as documents and items as words. Under this formulation, user preferences are modeled as probability distributions over latent topics, allowing each user – item interaction to be explained by different latent factors. This distributional representation enables LDA to capture heterogeneous user interests more flexibly than single-vector latent representations.

3.1.1.3   Content-based Filtering

Content-based filtering (CBF) recommends items to users by modeling user preferences from the attributes of items they have previously interacted with. Typically, both users and items are represented in a shared feature space, where recommendations are generated based on the similarity between user profiles and item representations. Since CBF relies solely on individual user history, it is less affected by user – user interaction sparsity but often suffers from limited diversity and difficulty in capturing evolving or complex user interests.

Salton et al. [20] proposed the vector space model for information retrieval, in which both documents and queries are represented as weighted term vectors. By assigning importance weights to terms (e.g., inverse document frequency) and computing similarity scores between query and document vectors, the model enables ranked retrieval based on relevance. This representation and similarity-matching paradigm laid the foundation for content-based recommender systems, where user profiles and item content are similarly modeled in a shared feature space.

Pazzani and Billsus [21] proposed a content-based recommender system that learns user profiles from explicit user feedback on item content. Their method represents items using content features and employs a naive Bayesian classifier to incrementally learn and revise user preference profiles, enabling the system to predict the interestingness of unseen items.

3.1.1.4   Hybrid Methods

Hybrid recommender systems combine both collaborative filtering and content-based filtering techniques to leverage the strengths of each approach and mitigate their respective weaknesses. By jointly exploiting user – item interaction patterns and item content information, hybrid methods can alleviate issues such as data sparsity and cold-start that commonly affect pure collaborative filtering models. These approaches typically integrate multiple signals at different stages of the recommendation pipeline, resulting in more robust and accurate predictions.

Burke [5] presented a comprehensive survey of hybrid recommender systems, systematically categorizing hybridization strategies such as weighted, switching, mixed, and feature combination approaches. The survey analyzed how different hybrid designs integrate multiple recommendation techniques to balance their respective strengths and weaknesses, demonstrating that hybrid methods can effectively improve recommendation accuracy and alleviate issues such as data sparsity and cold-start problems.

Pazzani [22] proposed a unified framework for integrating collaborative, content-based, and demographic filtering methods in recommender systems. By exploiting multiple sources of information, including user – item interactions, item content, and user profiles, the framework combines recommendations from different models to improve precision. Experimental results demonstrated that hybrid approaches within this framework consistently outperform single-method recommenders.

Melville et al. [23] proposed a content-boosted collaborative filtering (CBCF) framework that integrates content-based prediction into the collaborative filtering process. Specifically, a content-based predictor is first used to generate pseudo ratings for unrated items, producing a dense pseudo user – item matrix on which collaborative filtering is subsequently applied. By alleviating sparsity and the first-rater problem, this approach achieves significantly improved recommendation accuracy compared to pure collaborative, pure content-based, and naive hybrid methods.

Billsus et al. [24] developed a hybrid news recommender system for adaptive news access that integrates collaborative filtering and content-based filtering techniques. Their system learns personalized user models from both explicit and implicit user feedback, and combines short-term and long-term interest representations to adapt to users' evolving information needs. Deployed in a real-world news delivery environment, this work demonstrated the practical effectiveness of hybrid recommender systems in improving personalization quality without requiring additional user effort.

3.1.2   Supervised Machine Learning in Recommender Systems

Supervised machine learning techniques have also been widely applied to recommender systems to enhance recommendation accuracy. By formulating recommendation as a regression or classification problem, these methods learn predictive models from labeled user – item interaction data using features such as user demographics, item attributes, and contextual information. Algorithms including decision trees, support vector machines, and ensemble methods have been employed to capture complex relationships between features and user preferences. While effective in leveraging rich side information, supervised learning-based approaches often rely heavily on feature engineering and struggle to generalize under sparse interaction settings.

Basilico and Hofmann [25] proposed a unified supervised learning framework that integrates collaborative filtering and content-based filtering within a single prediction model. Their approach formulates recommendation as a learning problem over user – item pairs by designing joint feature representations

and kernel functions that enable simultaneous generalization across both user and item dimensions. By incorporating user – item interaction data together with item and user attributes, the framework achieves improved recommendation accuracy compared to traditional collaborative or content-based methods.

Rendle [26] introduced Factorization Machines (FMs), a supervised learning model that generalizes matrix factorization by modeling pairwise feature interactions through factorized parameters. By representing user – item interactions, item attributes, and contextual information as sparse feature vectors, FMs can efficiently capture interactions in high-dimensional and highly sparse settings. This unified formulation subsumes several state-of-the-art factorization models and has demonstrated superior performance over traditional collaborative and content-based approaches in various recommendation tasks.

Burges et al. [27] proposed a learning-to-rank framework that directly optimizes the ordering of items rather than predicting absolute preference scores. Their approach formulates ranking as a pairwise learning problem and introduces RankNet, which models ranking preferences using a probabilistic cost function optimized via gradient descent. By focusing on ranking quality, this framework significantly improves recommendation effectiveness in scenarios where the relative order of items is more important than precise rating prediction.

Later, Burges [28] provided a comprehensive overview of learning-to-rank methods, including RankNet, LambdaRank, and LambdaMART. RankNet formulates ranking as a pairwise probabilistic learning problem optimized via gradient descent, while LambdaRank introduces the concept of lambda gradients to directly optimize ranking metrics such as NDCG. By combining LambdaRank with gradient-boosted decision trees, LambdaMART further improves ranking performance and has become a widely adopted approach in large-scale recommendation and information retrieval systems.

In industrial recommender systems, supervised learning techniques are widely adopted due to their strong predictive performance and flexibility in incorporating heterogeneous features. Among these methods, decision tree-based models and ensemble learning techniques are particularly popular, as they provide a good balance between interpretability and the ability to capture complex feature interactions.

He et al. [29] developed a large-scale recommender system for Facebook Ads based on gradient boosting decision trees. By modeling user preferences from rich user, item, and contextual features, their approach significantly improved ad targeting effectiveness and user engagement, demonstrating the practicality of supervised learning methods in real-world industrial recommendation scenarios.

Furthermore, advanced gradient boosting frameworks such as XGBoost [30] and LightGBM [31] have been widely applied in recommender systems to enhance both accuracy and scalability. These methods leverage efficient tree-based boosting strategies to model high-order feature interactions, making them particularly suitable for large-scale recommendation tasks with sparse and high-dimensional feature spaces.

3.1.3   Deep Learning-based Recommender Systems

Deep learning-based recommender systems have significantly advanced the field by enabling end-to-end representation learning and modeling complex, non-linear user – item interactions. Neural Collaborative Filtering (NCF) extends traditional matrix factorization by replacing fixed inner products with multi-layer perceptrons, allowing the model to learn more expressive interaction functions. Beyond

interaction modeling, convolutional neural networks (CNNs) have been widely used to extract informative representations from unstructured item content such as images and text, thereby enriching item features for recommendation. Recurrent neural networks (RNNs) and their variants further incorporate temporal dynamics by modeling sequential user behaviors, enabling personalized recommendations that adapt to users' evolving preferences.

3.1.3.1 Neural Collaborative Filtering

Neural Collaborative Filtering (NCF) is a deep learning-based recommendation framework that replaces the fixed inner product used in matrix factorization with neural networks to model user – item interactions. By learning non-linear interaction functions through multi-layer perceptrons, NCF can capture more complex preference patterns than traditional collaborative filtering methods. The framework unifies several neural architectures, including generalized matrix factorization (GMF), multi-layer perceptron (MLP), and their hybrid variant NeuMF, which have demonstrated superior performance on various recommendation benchmarks.

He et al. [32] proposed the Neural Collaborative Filtering (NCF) framework, which formulates collaborative filtering as a neural interaction learning problem. Instead of relying on a fixed inner product as in traditional matrix factorization, NCF employs neural networks to learn flexible and non-linear user – item interaction functions directly from data. In this framework, users and items are embedded into low-dimensional latent spaces and their representations are combined through neural architectures—such as generalized matrix factorization (GMF), multi-layer perceptrons (MLP), and their fusion model NeuMF—to capture complex interaction patterns beyond linear similarity measures.

3.1.3.2 Sequential Recommender Systems

Sequential recommender systems exploit the sequential patterns in users' interaction histories to generate personalized recommendations. By explicitly modeling the order and temporal dependencies of user – item interactions, these methods capture the dynamic evolution of user preferences over time and have been widely applied in domains such as e-commerce, music streaming, and video platforms. Representative techniques for sequential recommendation include recurrent neural networks (RNNs) and their variants such as long short-term memory (LSTM) networks, as well as more recent Transformer-based architectures.

Hidasi et al. [33] proposed a session-based recommender system that applies recurrent neural networks (RNNs) to model user behavior within individual sessions. By representing a session as a sequence of item interactions and maintaining a recurrent hidden state, their approach captures both short-term and long-term dependencies in session data. Furthermore, the authors introduced ranking-oriented loss functions tailored to recommendation tasks, enabling the model to significantly outperform traditional item-to-item and neighborhood-based baselines.

The mainstream sequential recommender systems now are Transformer-based models. Kang and McAuley [34] proposed SASRec, a self-attentive sequential recommender system based on the Transformer encoder architecture. By employing self-attention mechanisms, SASRec adaptively weighs historical items in a user's interaction sequence, enabling the model to capture long-range dependencies while remaining efficient on sparse data. Unlike recurrent models that summarize sequences

through a single hidden state, SASRec directly attends to relevant past interactions, leading to improved recommendation accuracy and scalability in sequential recommendation tasks.

Sun et al. [35] proposed BERT4Rec, a Transformer-based sequential recommender system that employs bidirectional self-attention to model user behavior sequences. Unlike unidirectional sequential models such as RNN-based methods and SASRec, BERT4Rec leverages bidirectional contextual information by predicting masked items within a sequence using a Cloze-style training objective. This design enables each item representation to incorporate both preceding and succeeding context, leading to more expressive sequence modeling and consistently improved recommendation performance across multiple benchmark datasets.

3.1.3.3   Graph-based Recommender Systems

Graph-based recommender systems model user – item interactions as graphs and apply graph neural networks (GNNs) to learn representations through neighborhood aggregation. By propagating information along graph edges, these methods can effectively capture high-order connectivity and collaborative signals that are difficult to model with point-wise interaction functions. Moreover, graph-based frameworks naturally support the integration of side information and heterogeneous relations, enabling richer modeling of user preferences and item characteristics.

Van den Berg et al. [36] proposed Graph Convolutional Matrix Completion (GCMC), which formulates collaborative filtering as a link prediction problem on a bipartite user – item interaction graph. By employing a graph convolutional auto-encoder architecture, GCMC learns user and item representations through message passing on the interaction graph and reconstructs ratings via a bilinear decoder. This approach effectively captures high-order collaborative signals and naturally incorporates side information, leading to improved recommendation performance on benchmark datasets.

Ying et al. [37] proposed PinSage, a graph-based recommender system designed for web-scale applications. PinSage combines graph neural networks with efficient random-walk-based neighborhood sampling to learn item representations that incorporate both graph structure and rich side information. By addressing the scalability limitations of conventional GCNs, PinSage was successfully deployed in large-scale industrial systems such as Pinterest, demonstrating the effectiveness of graph-based recommendation models in real-world production environments.

Wang et al. [38] proposed Neural Graph Collaborative Filtering (NGCF), which explicitly integrates graph neural networks into collaborative filtering by modeling user – item interactions as a bipartite graph. NGCF refines user and item embeddings through recursive message passing on the interaction graph, enabling the explicit modeling of high-order connectivity and collaborative signals, which leads to significant improvements in recommendation performance.

He et al. [39] introduced LightGCN, a simplified graph convolutional network tailored for recommender systems. Unlike prior GNN-based models that incorporate feature transformations and nonlinear activations, LightGCN argues that these components contribute little to collaborative filtering performance when only user and item IDs are available as input. Accordingly, LightGCN retains only the neighborhood aggregation operation to propagate embeddings over the user – item interaction graph, significantly simplifying the model architecture while preserving the ability to capture high-order connectivity. Extensive experiments demonstrate that LightGCN not only reduces computational complexity but also achieves superior recommendation accuracy compared to more complex

GNN-based methods such as NGCF, making it a widely adopted and strong baseline in graph-based recommender system research.

Wu et al. [40] proposed SR-GNN, a session-based recommender system that models user interaction sequences as graph-structured data and applies graph neural networks to learn item representations within sessions. By constructing a directed session graph for each interaction sequence, SR-GNN is able to capture complex transition patterns among items that go beyond simple sequential dependencies. This work represents an early attempt to integrate graph-based modeling with sequential recommendation, effectively combining the strengths of GNNs in capturing high-order relational information and sequential models in characterizing short-term user intent.

3.1.3.4   Generative Recommender Systems

Generative recommender systems have emerged as an important research direction that models user preferences and item characteristics from a probabilistic perspective. By leveraging generative models such as Variational Autoencoders (VAEs), Generative Adversarial Networks (GANs), and Normalizing Flows, these approaches aim to learn the underlying distribution of user – item interactions rather than deterministic point estimates. Such distribution-aware modeling enables recommender systems to capture uncertainty in user preferences, generate diverse recommendations, and alleviate data sparsity issues. These advantages are closely related to the motivation of our proposed method, which also represents user preferences as probability distributions to support more expressive and robust recommendation.

Liang et al. [41] proposed a variational autoencoder (VAE)-based collaborative filtering framework for implicit feedback data. By modeling user preferences as latent random variables and employing a multinomial likelihood, this approach provides a probabilistic formulation that captures uncertainty and multi-modal structures in user behavior. Compared with deterministic autoencoder-based models, the VAE framework enables more expressive preference modeling and demonstrates strong empirical performance under sparse interaction settings.

Wang et al. [42] proposed IRGAN, a generative adversarial framework that unifies generative and discriminative models for information retrieval and recommendation tasks. By formulating the learning process as a minimax game, the generative model aims to approximate the underlying relevance distribution over items, while the discriminative model learns to distinguish relevant from non-relevant user – item pairs. Through adversarial training, IRGAN effectively improves recommendation performance, particularly under implicit feedback settings.

Chae et al. [43] proposed CFGAN, a generic collaborative filtering framework based on generative adversarial networks. Unlike prior GAN-based recommender systems that generate discrete item indices, CFGAN adopts vector-wise adversarial training, where the generator produces real-valued preference vectors and the discriminator distinguishes them from ground-truth interaction vectors. This design effectively stabilizes adversarial learning and improves recommendation accuracy, especially under sparse implicit feedback settings.

Beyond VAEs and GANs, diffusion models have recently been introduced to recommender systems as a new class of generative models. Wang et al. [44] proposed DiffRec, a diffusion-based recommender system that models the user – item interaction generation process through iterative denoising. By gradually corrupting user interaction histories and learning to recover the original interactions step by step, DiffRec provides a flexible and expressive framework for modeling complex preference dis-

tributions. This work demonstrates the potential of diffusion models to overcome the limitations of traditional generative approaches and further improve recommendation performance under noisy and sparse interaction settings.

3.2   Cross-Domain Recommender Systems

Despite the success of single-domain recommender systems, challenges such as data sparsity and cold-start users remain difficult to address when interaction data are limited. Cross-Domain Recommendation (CDR) tackles these issues by transferring knowledge from a source domain with abundant user – item interactions to a target domain where data are scarce.

The core objective of CDR is to leverage auxiliary information across domains to improve recommendation performance, under the assumption that user preferences or item characteristics exhibit certain transferable patterns. By exploiting correlations between domains, CDR methods aim to enhance recommendation accuracy and robustness, especially in cold-start and sparse-data scenarios.

Based on whether domains share common entities, existing CDR approaches can be broadly categorized into *overlapping* and *non-overlapping* settings. Overlapping CDR methods assume the existence of shared users or items across domains and utilize this overlap as a bridge for knowledge transfer in the training stage. Typical techniques include joint matrix factorization, co-clustering, and graph-based models that align user preferences or item representations across domains.

In contrast, non-overlapping CDR methods address more challenging scenarios where no users or items are shared between domains during training. These approaches generally rely on content features, latent representations, or learned mappings between domains to enable preference transfer without explicit entity overlap. Non-overlapping CDR is particularly relevant in practical applications, but remains challenging due to the lack of direct correspondence between domains.

3.2.1   Overlapping CDR

The core idea of overlapping CDR is to leverage the shared users or items between domains to facilitate knowledge transfer.

Singh and Gordon [45] proposed Collective Matrix Factorization (CMF), a multi-relational matrix factorization framework that jointly factorizes multiple user – item interaction matrices across domains. By sharing latent factors for overlapping users or items, CMF enables effective knowledge transfer between related domains and improves recommendation performance under data sparsity. This work is widely regarded as a foundational approach for overlapping cross-domain recommendation.

Li et al. [46] proposed a transfer learning framework for collaborative filtering based on a rating-matrix generative model. Their approach captures shared latent rating patterns across domains by learning a common cluster-level rating structure, enabling knowledge transfer even under severe data sparsity. This framework provides a principled probabilistic formulation for overlapping cross-domain recommendation and significantly improves recommendation performance in the target domain.

Pan et al. [47] proposed a transfer learning framework for collaborative filtering that alleviates data sparsity by transferring knowledge from auxiliary domains. Their method discovers shared latent structures of users and items in auxiliary data through matrix factorization and adapts these structures to the target domain via a principled regularization scheme. By exploiting overlapping users or

items across domains, this approach effectively captures transferable preference patterns and improves recommendation performance in sparse target domains.

Zhu et al. [48] proposed WCDR, a dual-target cross-domain recommendation framework that models users and items as probability distributions to explicitly capture uncertainty across domains. Unlike prior embedding-based overlapping CDR methods that rely on deterministic representations, WCDR represents user and item behaviors as elliptical Gaussian distributions and decouples them into local-intrinsic and global-domain components. To facilitate effective knowledge transfer between overlapping domains, WCDR further adopts a shared-specific paradigm for global-domain distributions and introduces a Masked Domain-aware Sub-distribution Aggregation (MDSA) module in the Wasserstein space. This design enables adaptive aggregation of domain-shared knowledge while alleviating negative transfer caused by domain imbalance. Extensive experiments demonstrate that modeling domain-level uncertainty and distributional relationships can significantly improve recommendation performance in both source and target domains. Their proposed method can be illustrated as in Figure 3.1.
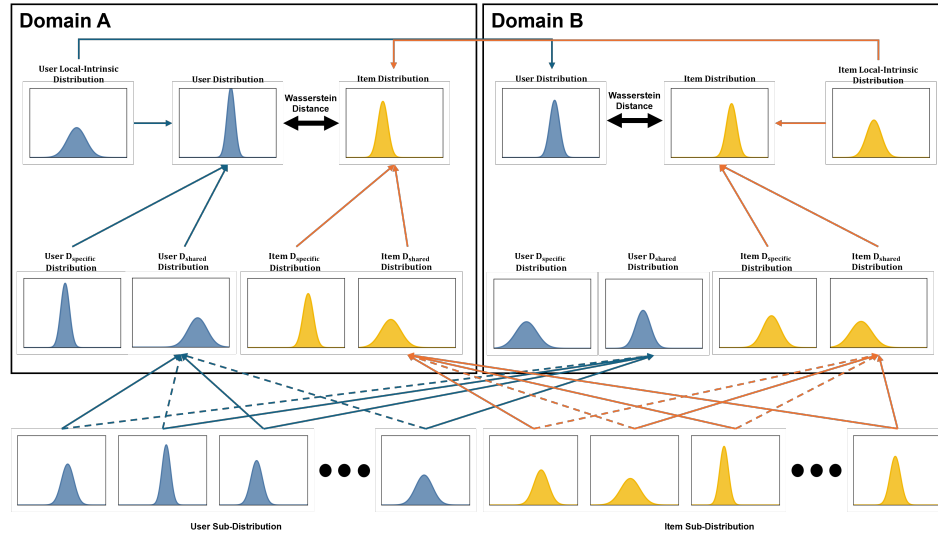

Figure 3.1: Illustration of the WCDR framework [48], which models users and items as probability distributions to capture uncertainty in overlapping cross-domain recommendation.

WCDR models users and items as probability distributions to capture uncertainty in overlapping cross-domain recommendation. By contrast, our work considers a more challenging non-overlapping CDR scenario without any shared users or items, where direct entity-level alignment is unavailable in the training stage and cross-domain transfer must rely on distribution-level modeling.

3.2.2   Non-overlapping CDR

Non-overlapping cross-domain recommendation addresses scenarios where no users or items are shared between domains. In the absence of explicit entity overlap, these methods aim to bridge domains

by learning transferable representations from auxiliary information, such as content features, latent factors, or deep neural models.

Man et al. [49] proposed EMCDR, an embedding-and-mapping framework for cross-domain recommendation. The framework first learns latent user and item representations independently in each domain, and then learns a cross-domain mapping function to project embeddings from the source domain into the target domain. By transferring preferences through latent space mapping, EMCDR enables effective knowledge transfer in non-overlapping and cold-start scenarios.

3.3　Distributional User Preference Modeling

Recent studies have recognized that user preferences are often uncertain and multi-modal, reflecting diverse and evolving interests. However, most traditional recommender systems represent user preferences as deterministic point embeddings in a latent space, which implicitly assume a single dominant preference pattern.

To better capture preference variability, distributional user preference modeling has been proposed as an alternative paradigm that represents user interests as probability distributions. By modeling preferences at the distribution level, these approaches provide a more expressive representation that can capture uncertainty, preference diversity, and complex user – item interaction patterns.

3.3.1　Multi-interest User Modeling

Multi-interest user modeling refers to a class of approaches that represent user preferences using multiple latent vectors or components in order to capture diverse and heterogeneous interests.

Early probabilistic models have implicitly adopted multi-interest representations by modeling user preferences as mixtures over latent components. Probabilistic Latent Semantic Analysis (pLSA) [17] and related latent class models represent users through distributions over latent topics or preference patterns inferred from interaction data. Latent Dirichlet Allocation (LDA) [19] further introduces a fully generative formulation by modeling user-specific mixture weights as random variables, enabling users to exhibit multiple interests to different degrees. Similarly, Marlin [18] models user rating profiles as mixtures of latent user attitudes, providing an early probabilistic formulation of multi-interest user preferences in collaborative filtering.

Zhou et al. [50] proposed the Deep Interest Network (DIN), which models users' diverse interests through a target-aware attention mechanism. Instead of compressing all historical behaviors into a fixed-length representation, DIN dynamically aggregates user behavior embeddings conditioned on the target item, allowing different interests to be activated for different recommendation candidates. This adaptive representation enables DIN to capture the multi-faceted nature of user preferences and has been shown to achieve strong performance in large-scale industrial recommender systems.

Li et al. [51] proposed the Multi-Interest Network with Dynamic Routing (MIND), which explicitly represents each user with multiple interest vectors. MIND employs a dynamic routing mechanism to cluster user behaviors into distinct interest representations, enabling the model to capture diverse and heterogeneous user preferences. By decoupling interest extraction from item matching, MIND is suitable for large-scale retrieval scenarios and has been successfully deployed in industrial recommender systems.

Cen et al. [52] proposed ComiRec, a controllable multi-interest recommender framework that explicitly models users with multiple interest representations. ComiRec employs capsule networks or self-attention mechanisms to extract diverse user interests from behavior sequences, and introduces an aggregation module to balance recommendation accuracy and diversity. By jointly considering multi-interest extraction and controllable aggregation, ComiRec extends prior multi-interest methods and has demonstrated strong effectiveness in large-scale industrial recommender systems.

3.3.2   Probabilistic Preference Modeling

Probabilistic preference modeling represents user preferences using probability distributions, providing a principled framework for characterizing uncertainty and variability in user behavior. Unlike multi-interest user modeling methods that describe preferences with a finite set of deterministic representations, probabilistic approaches explicitly model preference uncertainty and continuous variations in latent user interests. By treating user preferences as random variables, these methods offer a more expressive representation that is well suited for capturing diverse and evolving user behaviors.

Several early works, including the models proposed by Marlin [18], Hofmann [17], and Blei et al. [19], represent user preferences using probabilistic latent variable models, where user interests are characterized as distributions rather than deterministic embeddings. While these methods primarily focus on capturing the multi-faceted nature of user interests, probabilistic preference modeling also emphasizes uncertainty estimation, which becomes particularly important under sparse or noisy interaction data.

Salakhutdinov and Mnih [16] further advanced this direction by introducing Bayesian Probabilistic Matrix Factorization (BPMF), which models user and item latent factors as Gaussian distributions and performs Bayesian inference to explicitly account for uncertainty in preference estimation.

VAE-based methods provide another probabilistic framework for modeling uncertainty in user preferences. Liang et al. [41] extended variational autoencoders to collaborative filtering by modeling user preferences as latent random variables and performing variational inference. By representing users with posterior distributions in the latent space, VAE-based recommender systems are able to capture both the variability and uncertainty inherent in user behavior, leading to more robust recommendation performance under sparse interaction data.

Beyond recommender system – specific models, some studies investigate representing entities as probability distributions in general representation learning. Vilnis and McCallum [53] proposed Gaussian embeddings, which represent words as Gaussian distributions in the embedding space, enabling the modeling of both semantic uncertainty and asymmetric relationships. Although not originally designed for recommender systems, this distributional representation paradigm provides important methodological insights and has inspired subsequent work on modeling users and items as distributions in recommendation tasks.

3.4   Optimal Transport for Representation Alignment

Optimal Transport (OT) provides a principled framework for measuring and aligning probability distributions by computing an optimal transport plan that minimizes the cost of transforming one distribution into another. In recent years, OT has been increasingly adopted in machine learning as a

powerful tool for representation alignment, enabling comparisons and mappings between distributions in a geometrically meaningful way. This distribution-level alignment capability has led to successful applications of OT in domain adaptation, generative modeling, and representation learning.

3.4.1   OT Basics in ML

Optimal Transport is widely used in machine learning as a distribution-aware alignment technique. Its ability to compare probability distributions while respecting their geometric structure has led to successful applications in domain adaptation and representation learning.

Villani [54] systematically developed the theoretical foundations of optimal transport, formalizing the Kantorovich relaxation of the Monge problem and introducing Wasserstein distances as principled metrics between probability measures. By framing distribution comparison as a cost-minimizing mass transportation problem, this work established a rigorous mathematical basis for measuring and aligning probability distributions, which later enabled the adoption of OT as a core tool in machine learning tasks such as distribution alignment, domain adaptation, and representation learning.

Peyré and Cuturi [55] provided a comprehensive survey of optimal transport from a computational and machine learning perspective, systematically bridging OT theory with practical algorithms. Their work reviewed a wide range of OT-based applications, including domain adaptation, generative modeling, and deep learning, and emphasized scalable solutions such as entropic regularization and Sinkhorn iterations. By demonstrating how OT can be efficiently integrated into modern learning pipelines, this survey established OT as a practical and versatile tool for distribution comparison and representation alignment in machine learning.

Cuturi and Marco [56] introduced the Sinkhorn algorithm, which incorporates an entropic regularization term into the optimal transport formulation to obtain a smooth and strictly convex objective. This regularization allows the resulting OT problem to be solved efficiently via iterative matrix scaling, leading to orders-of-magnitude speedups compared to classical linear programming solvers. As a result, Sinkhorn-based OT distances make optimal transport computationally feasible for large-scale machine learning applications and have become a cornerstone for OT-based representation learning and distribution alignment methods.

Genevay et al. [57] proposed Sinkhorn divergences as a differentiable and computationally tractable OT-based loss for training generative models. By introducing entropic regularization and leveraging automatic differentiation through Sinkhorn iterations, their approach enables stable optimization and efficient gradient computation at scale. This work demonstrated that OT-based losses can effectively improve the quality of generated samples, highlighting the potential of optimal transport as a powerful tool for generative modeling.

3.4.2   OT in Domain Adaptation

Optimal Transport has been extensively studied in domain adaptation as a principled framework for aligning feature distributions between source and target domains. By explicitly modeling the discrepancy between distributions and minimizing the cost of transporting probability mass across domains, OT-based methods enable effective knowledge transfer from the source domain to the target domain. As a result, these approaches can reduce domain shift and improve model performance in the target domain, especially when the two domains exhibit substantial distributional differences.

Courty et al. [58] proposed a seminal optimal-transport-based framework for unsupervised domain adaptation, in which feature distributions from the source and target domains are aligned by learning an optimal transportation plan. By minimizing the Wasserstein distance between empirical distributions, their method explicitly addresses domain shift at the distribution level and enables effective knowledge transfer from the source domain to the target domain. Moreover, the proposed regularized OT formulation allows the incorporation of class and structural information, leading to improved adaptation performance and establishing OT as a principled tool for representation alignment in domain adaptation.

Frogner et al. [59] introduced a deep learning framework that incorporates the Wasserstein distance as a loss function for representation learning and domain adaptation. By embedding OT-based losses into neural networks, their method enables end-to-end training while explicitly accounting for the geometric structure of the output space. Through an entropically regularized formulation, the proposed Wasserstein loss can be efficiently optimized via gradient-based methods, allowing effective alignment of feature distributions across domains and demonstrating the feasibility of OT within deep learning frameworks.

With the development of efficient OT solvers such as the Sinkhorn algorithm, OT-based domain adaptation methods have become computationally tractable for large-scale applications. These advances significantly improve the practical applicability of OT, and extensive empirical results have shown that OT-based approaches can effectively mitigate distributional shifts between source and target domains, leading to consistent performance gains across a wide range of domain adaptation tasks.

Chapter 4

Distributional Preference Modeling for Cross-Domain Recommendation

4.1   Background

Recommender systems have achieved remarkable success in a wide range of real-world applications, such as e-commerce, online media platforms, and social networks. However, despite their effectiveness, many existing recommender systems continue to face fundamental challenges, including data sparsity and the cold-start problem, particularly when user – item interactions are limited or unevenly distributed. These challenges often lead to suboptimal recommendation performance, as traditional models struggle to accurately capture user preferences under such constraints.

Most traditional recommender systems model user preferences as fixed point embeddings in a latent space. While such representations are computationally efficient, they implicitly assume that a user's interests can be captured by a single deterministic vector. This assumption overlooks two important characteristics of real user behavior. First, user preferences are inherently uncertain, especially in sparse settings where limited observations are available. Second, user interests are often multi-faceted, spanning multiple latent aspects that cannot be adequately represented by a single point estimate. As a result, point-based representations may fail to capture the diversity and ambiguity of user preferences, leading to suboptimal recommendation performance.

Cross-domain recommendation (CDR) aims to alleviate cold-start and data sparsity issues by transferring knowledge from a data-rich source domain to a data-sparse target domain. A common strategy adopted by existing CDR methods is to exploit overlapping users or items between domains as explicit bridges for knowledge transfer. Through shared embeddings, adversarial alignment, or joint training objectives, these methods assume that at least part of the user or item space is shared across domains during training. However, such assumptions are often unrealistic in practical scenarios. In many real-world applications, user identities or item catalogs are domain-specific due to privacy constraints, platform isolation, or delayed synchronization, resulting in entirely non-overlapping users and items across domains during the training phase.

In the absence of overlapping entities, effective knowledge transfer across domains becomes sub-

stantially more challenging. Without explicit correspondences, aligning latent representations across domains requires modeling higher-level structural or distributional similarities rather than instance-level matches. Existing non-overlapping CDR methods typically rely on shared latent spaces or distribution matching techniques, but most of them still represent user preferences as discrete vectors, limiting their expressiveness and robustness under severe data sparsity.

Motivated by these limitations, we propose to model user preferences from a distributional perspective and perform cross-domain knowledge transfer at the distribution level. Specifically, we introduce **DUP-OT** (Distributional User Preference with Optimal Transport), a novel framework for cross-domain recommendation under strictly non-overlapping settings. The core idea of DUP-OT is to represent each user's preference as a probability distribution over latent preference components, rather than a single point embedding. This distributional representation explicitly captures both the uncertainty and the multi-aspect nature of user interests.

To enable effective cross-domain alignment between such distributional preferences, DUP-OT leverages optimal transport (OT) theory as a principled mechanism for measuring and aligning probability distributions across domains. By aligning distributional representations in a shared latent space, OT allows preference knowledge to be transferred from the source domain to the target domain without requiring any overlapping users or items during training. This design enables DUP-OT to bridge domain gaps at a structural level, making it particularly suitable for realistic cross-domain recommendation scenarios where explicit correspondences are unavailable.

In the following sections, we introduce the detailed architecture of DUP-OT, including its shared preprocessing stage for constructing a unified latent space, the distributional user preference modeling module based on Gaussian Mixture Models, and the optimal-transport-based alignment mechanism for cross-domain preference transfer.

4.2   Methods

4.2.1   Overview

The overall architecture of the proposed DUP-OT framework is illustrated in Figure 4.1. DUP-OT is designed for cross-domain recommendation under strictly non-overlapping settings and consists of three main stages: (1) Shared Preprocessing Stage, (2) User GMM Weights Learning Stage, and (3) Cross-Domain Rating Prediction Stage. These stages jointly enable distributional user preference modeling and cross-domain knowledge transfer via optimal transport.

The scenario setup of DUP-OT involves two domains: a source domain $\mathcal{D}_S$ and a target domain $\mathcal{D}_T$. Each domain contains its own set of users and items, with **no overlapping users or items** between the two domains during training. The training set of source domain should happen before the valid and test set of target domain in time to avoid information leakage. The goal of DUP-OT is to leverage the abundant user – item interaction data in the source domain to enhance recommendation performance in the target domain, particularly under data sparsity conditions.

A core design principle of DUP-OT is to model user preferences as probability distributions rather than point embeddings. Specifically, we represent each user's preference as a Gaussian Mixture Model (GMM) in a shared latent space across both source and target domains. To make distribution-level alignment computationally feasible, we introduce the following assumption: within each domain, all
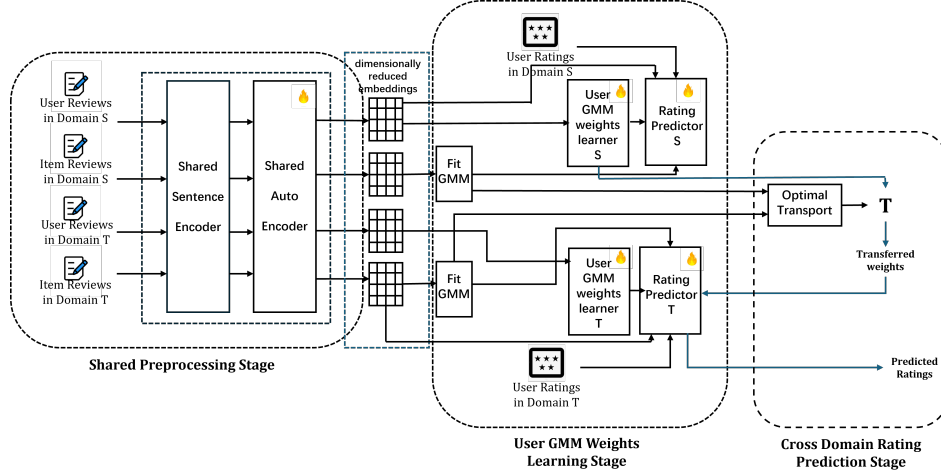
Figure 4.1: Architecture of the DUP-OT Framework

users share a fixed set of GMM components, while only the mixture weights vary across users. The shared components capture the main latent preference aspects of the domain, whereas the user-specific weights reflect individual preferences over these aspects. This assumption is reasonable in practice, as users within the same domain often exhibit similar underlying preference structures, and it allows us to perform cross-domain alignment at the component level instead of the instance level.

The Shared Preprocessing Stage aims to construct a unified latent space for both domains. Given user − item interaction data accompanied by review texts, ratings, and timestamps, we first extract semantic features from review texts using a shared pre-trained BERT model. User and item embeddings are obtained by aggregating review-level features, where a time-decay function is applied to assign higher weights to more recent reviews during user embedding aggregation. As for item embeddings, we simply average all associated review embeddings. This preprocessing pipeline is shared across domains to ensure consistency in representation. Since the resulting embeddings are high-dimensional, a shared autoencoder is trained on data from both domains to reduce dimensionality and produce compact embeddings in a common latent space, which serves as the basis for subsequent preference modeling.

Based on the reduced item embeddings, we then determine the fixed GMM components for each domain. Specifically, a Gaussian Mixture Model is fitted to all item embeddings within a domain, and the learned mixture components are treated as domain-level latent preference aspects. This design is motivated by the observation that items in a domain naturally reflect its major semantic and preference dimensions.

In the User GMM Weights Learning Stage, DUP-OT learns personalized preference distributions for individual users. For each domain, we train a user-specific GMM weight learner and a rating prediction model using only data from that domain. The weight learner, implemented as a multi-layer perceptron (MLP), maps a user's reduced embedding to a set of mixture weights over the fixed GMM components. These weights define the user's preference distribution. A separate rating prediction

23

MLP then estimates user – item ratings based on the weighted Mahalanobis distances between item embeddings and the GMM components. This stage produces expressive distributional representations of user preferences within each domain.

Finally, the Cross-Domain Rating Prediction Stage enables knowledge transfer from the source domain to the target domain via optimal transport. Since GMM components are fixed within each domain, we formulate optimal transport at the component level to align source-domain and target-domain GMMs. The resulting transport plan specifies how preference mass should be transferred between components across domains. Using this transport plan, user-specific GMM weights learned in the source domain can be mapped to the target domain, yielding adapted preference distributions. These transferred distributions are optionally fused with original target-domain user distributions to enhance preference modeling. The final user preference distributions are then used by the target-domain rating prediction model to generate improved rating predictions.

4.2.2   Shared Preprocessing Stage

The Shared Preprocessing Stage aims to extract consistent user and item embeddings from raw review data across both source and target domains. This stage consists of three main steps: (1) Review Text Embedding, (2) User and Item Embedding Aggregation, and (3) Dimensionality Reduction via Autoencoder. As our settings involve two domains with entirely non-overlapping users and items, it is crucial to ensure that the extracted embeddings are comparable and lie in a shared latent space. Also, the potential connections between the two domains can only be established through semantic similarities, so leveraging review texts is essential for capturing meaningful representations.

4.2.2.1   Review Text Embedding

To extract semantic features from review texts, we utilize a shared pre-trained BERT model across both domains. Specifically, we are using all-MiniMLM-L6-v2 model from Sentence-Transformers library [60], which is a lightweight variant of BERT optimized for generating sentence embeddings. Given a review text, we tokenize it and feed it into the BERT model to obtain a fixed-length embedding vector that captures its semantic content. By using a shared BERT model, we ensure that review embeddings from both domains are generated in the same semantic space, facilitating cross-domain alignment later on.

4.2.2.2   User and Item Embedding Aggregation

Usually, the recent reviews of a user are more indicative of its current preferences or characteristics. To account for this temporal aspect, we apply a time-decay function when aggregating review embeddings into user embeddings. For each user, we aggregate review-level embeddings into an initial user representation using a time-aware weighted pooling strategy.

Given a set of reviews with timestamps $\{t_i\}_{i=1}^N$ and corresponding embeddings $\{\mathbf{e}_i\}_{i=1}^N$, we define the reference time as the most recent review timestamp $t_{\text{ref}} = \max_i t_i$. The temporal distance of each review is measured in months and truncated by a maximum value :

$$\Delta_i = \min\left(\frac{t_{\text{ref}} - t_i}{T}, \Delta_{\max}\right),$$

where $T = 30 \times 86400$. Each review is assigned an exponentially decayed weight:

$$w_i = \exp(-\lambda \Delta_i),$$

which is further normalized as $\tilde{w}_i = w_i / \sum_j w_j$. The final user embedding is computed as a weighted sum of review embeddings:

$$\mathbf{u} = \sum_{i=1}^{N} \tilde{w}_i \mathbf{e}_i.$$

For item embedding aggregation, we assume that item attributes are relatively stable over time. Therefore, we simply compute the item representation by averaging all associated review embeddings. Formally, the item embedding is computed as

$$\mathbf{v} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{e}_i,$$

where $\{\mathbf{e}_i\}$ are the embeddings of reviews associated with the item.

An example of item embeddings after aggregation is illustrated in Figure 4.2. It can be observed that item embeddings from different domains exhibit distinct distributional patterns, indicating a substantial domain discrepancy. Addressing this discrepancy is critical for effective cross-domain recommendation and constitutes the main focus of the following sections.
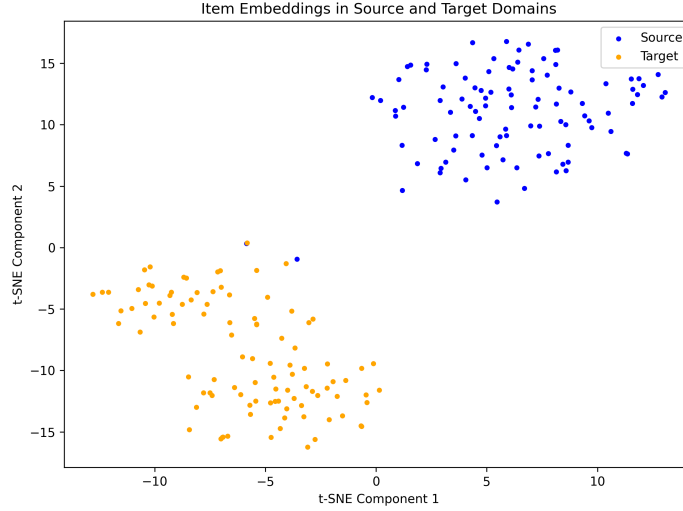

Figure 4.2: Item Embeddings after Aggregation

4.2.2.3   Dimensionality Reduction via Autoencoder

The aggregated user and item embeddings are typically high-dimensional, which can lead to increased computational costs and potential overfitting in subsequent modeling stages. To address this issue, we employ a shared autoencoder to reduce the dimensionality of embeddings from both domains. The autoencoder consists of an encoder network that maps input embeddings to a lower-dimensional latent space and a decoder network that reconstructs the original embeddings from the latent representations. The structure of the autoencoder is illustrated in Figure 4.3.
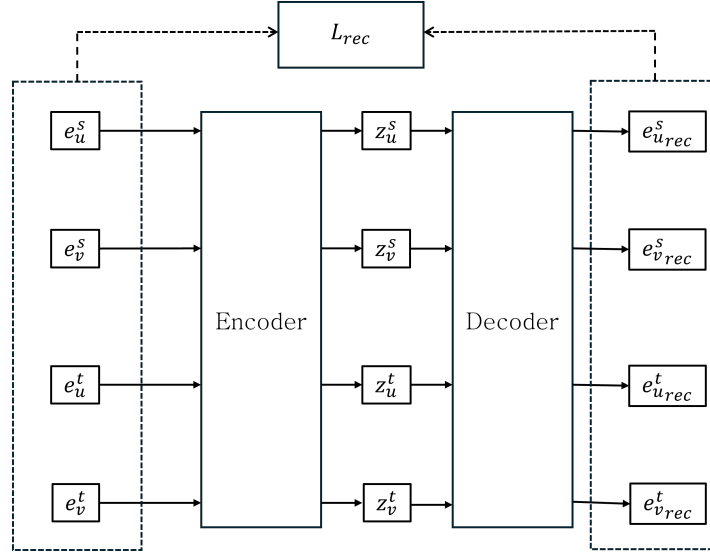

Figure 4.3: Autoencoder Structure

The autoencoder is trained on a combined dataset of user and item embeddings from both source and target domains. The training objective is to minimize the reconstruction loss, defined as the mean squared error between the original embeddings and their reconstructions. By sharing the autoencoder across domains, we ensure that the resulting reduced embeddings lie in a common latent space, which is essential for following distributional preference modeling and cross-domain alignment.

After training, we obtain reduced user and item embeddings by passing the original embeddings through the encoder network. These reduced embeddings serve as the basis for subsequent GMM-based preference modeling and optimal transport alignment in the DUP-OT framework.

4.2.3   User GMM Weights Learning Stage

In the User GMM Weights Learning Stage, we aim to learn expressive distributional representations of user preferences in the shared latent space constructed by the Shared Preprocessing Stage. Specifically, each user's preference is modeled as a Gaussian Mixture Model (GMM), which enables the representation of multi-aspect user interests in a probabilistic manner. This stage consists of two key components:

(1) determining a fixed set of domain-level GMM components, and (2) learning user-specific mixture weights together with a corresponding rating prediction model.

4.2.3.1   Fixed GMM Component Determination

To capture the major latent preference aspects within each domain, we first construct a domain-level Gaussian Mixture Model (GMM) by fitting it to the reduced item embeddings obtained from the Shared Preprocessing Stage. The resulting GMM components are treated as fixed and shared by all users within the same domain, serving as latent preference components for subsequent user-specific preference modeling.

To build intuition for Gaussian Mixture Models, we first present a one-dimensional (1D) example in Figure 4.4, where the roles of individual Gaussian components and mixture weights can be easily visualized. We then illustrate a two-dimensional (2D) GMM in Figure 4.5 to demonstrate how mixture models generalize from scalar to vector-valued representations.
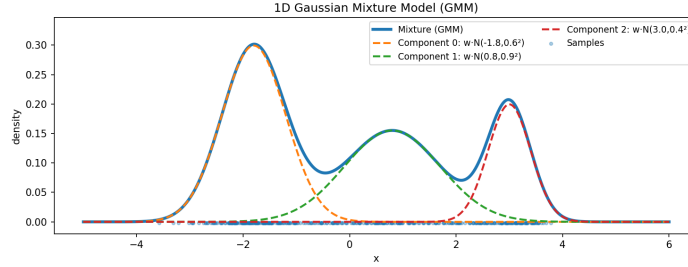

Figure 4.4: 1D Gaussian Mixture Model Example

To this end, we adopt the `BayesianGaussianMixture` implementation from the scikit-learn library [61] to fit GMMs on item embeddings. Unlike conventional GMMs that require manually specifying the number of mixture components, this approach is based on variational Bayesian inference with a Dirichlet prior over mixture weights. Such a formulation allows mixture components that are not sufficiently supported by the data to be automatically suppressed during training, enabling the effective number of active components to be inferred directly from the data.

Formally, given a set of item embeddings $\{\mathbf{x}_n\}_{n=1}^N$, the Bayesian Gaussian Mixture Model assumes the following generative process:

$$\boldsymbol{\pi} \sim \text{Dirichlet}(\boldsymbol{\alpha}), \qquad z_n \sim \text{Categorical}(\boldsymbol{\pi}), \qquad \mathbf{x}_n \mid z_n = k \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$

where $\boldsymbol{\pi}$ denotes the mixture weights, z_n is the latent component assignment, and $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ represent the mean and covariance of the k-th Gaussian component.

Exact posterior inference in this model is intractable. Therefore, variational Bayesian inference is employed by maximizing the evidence lower bound (ELBO):

$$\mathcal{L}(q) = \mathbb{E}_q[\log p(\mathbf{X}, \mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma})] - \mathbb{E}_q[\log q(\mathbf{Z}, \boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma})],$$

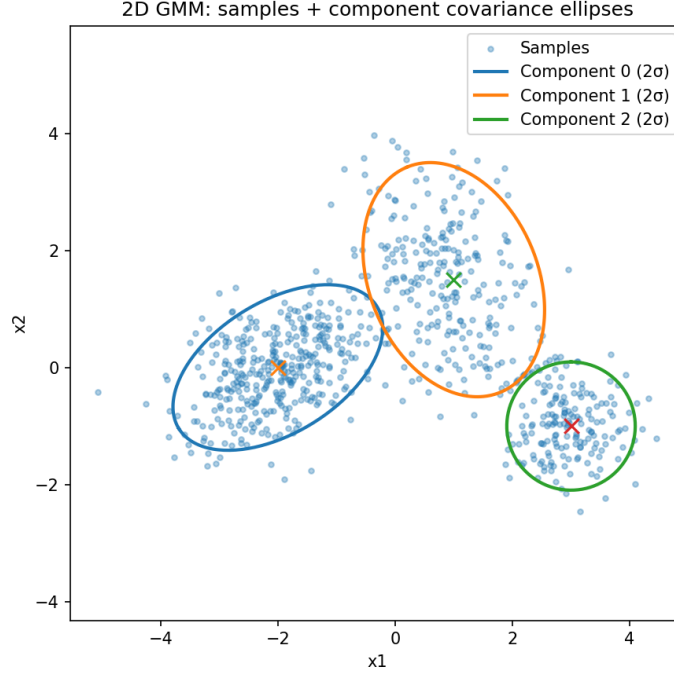where $q(\cdot)$ denotes the variational posterior distribution.

27

Figure 4.5: 2D Gaussian Mixture Model Example

Due to the Dirichlet prior imposed on the mixture weights, components that are weakly supported by the data are assigned negligible posterior mass, effectively pruning redundant components. As a result, the Bayesian GMM provides a principled and flexible mechanism for determining the effective number of domain-level latent preference components.

4.2.3.2   User-Specific GMM Weight Learning and Rating Prediction

With the fixed GMM components determined for each domain, we proceed to learn user-specific mixture weights that characterize individual preference distributions in the shared latent space. For each domain, DUP-OT employs two neural modules trained solely on the data from that domain: a user-specific GMM weight learner and a rating prediction network. These two modules are optimized jointly to ensure that the learned preference distributions are directly aligned with the rating prediction objective.

User-specific GMM weight learning. Let $\mathbf{u}_i \in \mathbb{R}^d$ denote the reduced embedding of user i obtained from the Shared Preprocessing Stage. Given the fixed set of domain-level GMM components $\{(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$, we learn personalized mixture weights via a multi-layer perceptron (MLP):

$$\mathbf{s}_i = f_\theta(\mathbf{u}_i) \in \mathbb{R}^K, \qquad \boldsymbol{\pi}_i = \mathrm{softmax}(\mathbf{s}_i),$$

where $\boldsymbol{\pi}_i = [\pi_{i1}, \ldots, \pi_{iK}]^\top$ represents the user-specific mixture weights satisfying $\sum_{k=1}^{K} \pi_{ik} = 1$ and $\pi_{ik} \geq 0$. The resulting weights define a personalized preference distribution over the domain-level latent preference components.

Distribution-aware user–item representation.   For an item j with reduced embedding $\mathbf{v}_j \in \mathbb{R}^d$, we compute its Mahalanobis distance to each GMM component:

$$d_{ijk} = \sqrt{(\mathbf{v}_j - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{v}_j - \boldsymbol{\mu}_k)}.$$

These component-wise distances are combined with the user-specific mixture weights to construct a distribution-aware representation of user–item compatibility:

$$\mathbf{h}_{ij} = [\pi_{i1}d_{ij1}, \ \pi_{i2}d_{ij2}, \ \ldots, \ \pi_{iK}d_{ijK}]^\top \in \mathbb{R}^K.$$

This formulation enables user preferences and item representations to be compared in a distribution-aware manner by jointly considering latent preference components and their personalized importance.

Rating prediction and joint training.   A separate MLP is employed to predict the rating for a given user–item pair:

$$\hat{r}_{ij} = g_\phi(\mathbf{h}_{ij}),$$

where $g_\phi(\cdot)$ denotes the rating prediction network. The GMM weight learner f_θ and the rating prediction network g_ϕ are trained jointly in a supervised fashion by minimizing the rating prediction loss over the observed interactions $\mathcal{D}$:

$$\mathcal{L}_{\text{rate}}(\theta, \phi) = \frac{1}{|\mathcal{D}|} \sum_{(i,j,r_{ij}) \in \mathcal{D}} (\hat{r}_{ij} - r_{ij})^2,$$

where r_{ij} and $\hat{r}_{ij}$ denote the ground-truth and predicted ratings, respectively. Through joint optimization, the learned user-specific preference distributions are directly tailored to the recommendation task.

4.2.4   Cross-Domain Rating Prediction Stage

In the Cross-Domain Rating Prediction Stage, we leverage optimal transport (OT) theory to facilitate knowledge transfer from the source domain to the target domain at the distribution level. It is important to emphasize that the non-overlapping constraint in our setting is imposed strictly during the training stage: no overlapping users or items are exploited when learning user preference models or cross-domain alignments. During inference and evaluation, however, users may appear in both domains, which reflects realistic deployment scenarios and allows transferred knowledge to be utilized in a principled manner. This design is reasonable because, in many real-world applications, cross-domain user correspondences are unavailable or restricted during training due to privacy concerns, system isolation, or data access limitations, while such correspondences may become observable at deployment or evaluation time.

Based on the User GMM Weights Learning Stage, each user's preference in both the source and target domains is represented as a Gaussian Mixture Model (GMM) defined over fixed, domain-level

latent preference components. Given these distributional representations, we formulate cross-domain knowledge transfer as an optimal transport problem between the source-domain and target-domain GMM component sets. Since all users within a domain share the same set of GMM components, OT is performed at the component level rather than the instance level, enabling efficient and stable alignment of latent preference aspects across domains.

Optimal transport formulation and cost matrix.   Formally, let $\boldsymbol{\pi}^s \in \mathbb{R}_+^{K_s}$ and $\boldsymbol{\pi}^t \in \mathbb{R}_+^{K_t}$ denote the mixture weight vectors over the fixed GMM components in the source and target domains, respectively. We define a cost matrix $\mathbf{C} \in \mathbb{R}^{K_s \times K_t}$, where each entry C_{kl} measures the dissimilarity between the k-th source-domain component and the l-th target-domain component.

Specifically, the cost matrix is constructed using the 2-Wasserstein distance between Gaussian distributions. For the k-th source-domain component $\mathcal{N}(\boldsymbol{\mu}_k^s, \boldsymbol{\Sigma}_k^s)$ and the l-th target-domain component $\mathcal{N}(\boldsymbol{\mu}_l^t, \boldsymbol{\Sigma}_l^t)$, the cost is defined as:

$$C_{kl} = \|\boldsymbol{\mu}_k^s - \boldsymbol{\mu}_l^t\|_2^2 + \mathrm{Tr}\left(\boldsymbol{\Sigma}_k^s + \boldsymbol{\Sigma}_l^t - 2\left((\boldsymbol{\Sigma}_k^s)^{1/2}\boldsymbol{\Sigma}_l^t(\boldsymbol{\Sigma}_k^s)^{1/2}\right)^{1/2}\right).$$

Figure 4.6 provides an intuitive illustration of how the cost matrix is constructed by computing pairwise Wasserstein distances between GMM components in the source and target domains. Each entry in the cost matrix corresponds to the transport cost between a pair of latent preference components, serving as the fundamental unit for subsequent OT alignment.
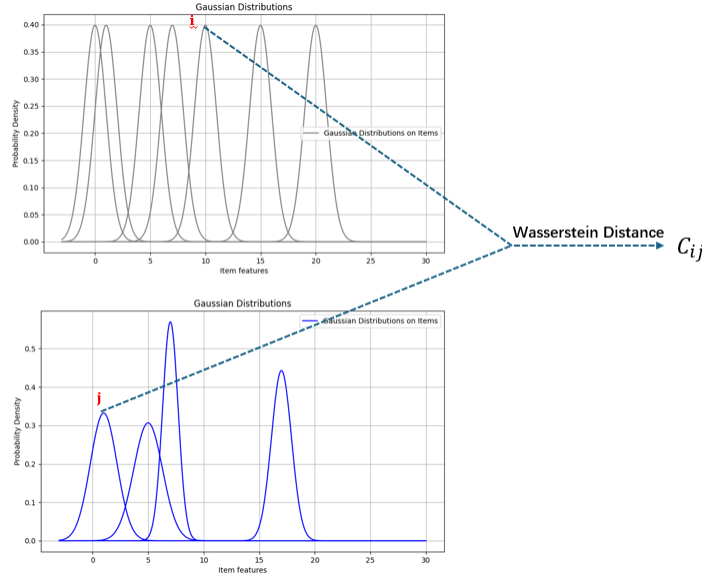

Figure 4.6: Cost Matrix Construction via Pairwise Wasserstein Distances

Given the cost matrix, the optimal transport plan $\mathbf{T}$ is obtained by solving:

$$\min_{\mathbf{T} \geq 0} \langle \mathbf{T}, \mathbf{C} \rangle \quad \text{s.t.} \quad \mathbf{T}\mathbf{1} = \boldsymbol{\pi}^s, \ \mathbf{T}^\top \mathbf{1} = \boldsymbol{\pi}^t.$$

When no regularization is applied, this formulation corresponds to the Earth Mover's Distance (EMD), which can be solved as a linear programming problem to obtain an exact and typically sparse transport plan. Alternatively, we adopt the entropically regularized OT formulation:

$$\min_{\mathbf{T} \geq 0} \langle \mathbf{T}, \mathbf{C} \rangle + \varepsilon \sum_{k,l} T_{kl}(\log T_{kl} - 1),$$

which can be efficiently solved using the Sinkhorn algorithm. The entropy regularization yields a smooth and dense transport plan while significantly reducing computational cost. In our setting, OT is performed at the component level, where the number of GMM components is relatively small. Therefore, both EMD and Sinkhorn-based OT are computationally feasible. In practice, we primarily adopt the Sinkhorn algorithm for efficiency and numerical stability, while EMD is used for analysis when an exact transport plan is desired.

Distribution transfer and fusion. Using the learned optimal transport plan, we transfer user-specific mixture weights from the source domain to the target domain as:

$$\tilde{\boldsymbol{\pi}}_i^t = \mathbf{T}^\top \boldsymbol{\pi}_i^s.$$

At inference time, we construct the final target-domain preference distribution via a linear fusion strategy:

$$\boldsymbol{\pi}_i^{\text{final}} = \lambda_i \boldsymbol{\pi}_i^t + (1 - \lambda_i)\tilde{\boldsymbol{\pi}}_i^t,$$

where $\lambda_i \in [0, 1]$ controls the contribution of the original target-domain distribution. If a user has interactions in both domains, $\lambda_i \in (0, 1)$ is used to combine domain-specific evidence with transferred knowledge. If a user has interactions only in the source domain, $\lambda_i = 0$ and the final distribution relies solely on the transferred preference. If a user has interactions only in the target domain, $\lambda_i = 1$ and the original target-domain distribution is retained.

Finally, the fused user preference distributions are fed into the target-domain rating prediction model to generate enhanced rating predictions. Through this training-time non-overlapping and inference-time adaptive fusion design, DUP-OT faithfully reflects practical cross-domain recommendation scenarios while maintaining a strict non-overlapping assumption during model learning.

Chapter 5

Experiments

Chapter 6

Results

Chapter 7

Conclusion

References

[1] D. Bawden and L. Robinson. Information overload: An overview. In *Oxford Encyclopedia of Political Decision Making*. Oxford University Press, Oxford, June 2020. This is a draft of a chapter that has been published by Oxford University Press in the book Oxford Encyclopedia of Political Decision Making.

[2] Jun Lv and Xuan Liu. The impact of information overload of e-commerce platform on consumer return intention: Considering the moderating role of perceived environmental effectiveness. *International Journal of Environmental Research and Public Health*, Vol. 19, No. 13, 2022.

[3] Guihua Zhang, Junwei Cao, and Dong Liu. Examining the influence of information overload on consumers' purchase in live streaming: A heuristic-systematic model perspective. *PLOS ONE*, Vol. 18, No. 8, pp. 1–16, 08 2023.

[4] Paul Resnick and Hal R Varian. Recommender systems. *Communications of the ACM*, Vol. 40, No. 3, pp. 56–58, 1997.

[5] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, Vol. 12, No. 4, pp. 331–370, 2002.

[6] Robin Burke, Alexander Felfernig, and Mehmet H. Göker. Recommender systems: An overview. *AI Magazine*, Vol. 32, No. 3, pp. 13–18, Jun. 2011.

[7] Dietmar Jannach and Michael Jugovac. Measuring the business value of recommender systems. *ACM Trans. Manage. Inf. Syst.*, Vol. 10, No. 4, December 2019.

[8] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez. Recommender systems survey. *Knowledge-Based Systems*, Vol. 46, pp. 109–132, 2013.

[9] Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, Vol. 2009, No. 1, p. 421425, 2009.

[10] Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, CSCW '94, p. 175–186, New York, NY, USA, 1994. Association for Computing Machinery.

[11] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, pp. 285–295, 2001.

[12] John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering, 2013.

[13] Lyle H Ungar and Dean P Foster. Clustering methods for collaborative filtering. In *AAAI workshop on recommendation systems*, Vol. 1, pp. 114–129. Menlo Park, CA, 1998.

[14] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, Vol. 42, No. 8, pp. 30–37, 2009.

[15] Andriy Mnih and Russ R Salakhutdinov. Probabilistic matrix factorization. In J. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems*, Vol. 20. Curran Associates, Inc., 2007.

[16] Ruslan Salakhutdinov and Andriy Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, p. 880–887, New York, NY, USA, 2008. Association for Computing Machinery.

[17] Thomas Hofmann. Latent semantic models for collaborative filtering. *ACM Trans. Inf. Syst.*, Vol. 22, No. 1, p. 89–115, January 2004.

[18] Benjamin M Marlin. Modeling user rating profiles for collaborative filtering. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems*, Vol. 16. MIT Press, 2003.

[19] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, Vol. 3, No. Jan, pp. 993–1022, 2003.

[20] Gerard Salton, Chris Buckley, and Edward A Fox. Automatic query formulations in information retrieval. *Journal of the American society for information science*, Vol. 34, No. 4, pp. 262–280, 1983.

[21] Michael Pazzani and Daniel Billsus. Learning and revising user profiles: The identification of interesting web sites. *Machine learning*, Vol. 27, No. 3, pp. 313–331, 1997.

[22] Michael J Pazzani. A framework for collaborative, content-based and demographic filtering. *Artificial intelligence review*, Vol. 13, No. 5, pp. 393–408, 1999.

[23] Prem Melville, Raymond J Mooney, Ramadass Nagarajan, et al. Content-boosted collaborative filtering for improved recommendations. *Aaai/iaai*, Vol. 23, pp. 187–192, 2002.

[24] Daniel Billsus and Michael J Pazzani. User modeling for adaptive news access. *User modeling and user-adapted interaction*, Vol. 10, No. 2, pp. 147–180, 2000.

[25] Justin Basilico and Thomas Hofmann. Unifying collaborative and content-based filtering. In *Proceedings of the twenty-first international conference on Machine learning*, p. 9, 2004.

[26] Steffen Rendle. Factorization machines. In *2010 IEEE International conference on data mining*, pp. 995–1000. IEEE, 2010.

[27] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pp. 89–96, 2005.

[28] Christopher JC Burges. From ranknet to lambdarank to lambdamart: An overview. *Learning*, Vol. 11, No. 23-581, p. 81, 2010.

[29] Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, et al. Practical lessons from predicting clicks on ads at facebook. In *Proceedings of the eighth international workshop on data mining for online advertising*, pp. 1–9, 2014.

[30] Tianqi Chen. Xgboost: A scalable tree boosting system. *Cornell University*, 2016.

[31] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, Vol. 30, , 2017.

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

[33] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. Session-based recommendations with recurrent neural networks. *arXiv preprint arXiv:1511.06939*, 2015.

[34] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pp. 197–206. IEEE, 2018.

[35] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, p. 1441–1450, New York, NY, USA, 2019. Association for Computing Machinery.

[36] Rianne Van Den Berg, N Kipf Thomas, and Max Welling. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*, Vol. 2, No. 8, p. 9, 2017.

[37] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 974–983, 2018.

[38] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, p. 165–174, New York, NY, USA, 2019. Association for Computing Machinery.

[39] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, YongDong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, p. 639–648, New York, NY, USA, 2020. Association for Computing Machinery.

[40] Shu Wu, Yuyuan Tang, Yanqiao Zhu, Liang Wang, Xing Xie, and Tieniu Tan. Session-based recommendation with graph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33, No. 01, pp. 346–353, Jul. 2019.

[41] Dawen Liang, Rahul G. Krishnan, Matthew D. Hoffman, and Tony Jebara. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, p. 689–698, Republic and Canton of Geneva, CHE, 2018. International World Wide Web Conferences Steering Committee.

[42] Jun Wang, Lantao Yu, Weinan Zhang, Yu Gong, Yinghui Xu, Benyou Wang, Peng Zhang, and Dell Zhang. Irgan: A minimax game for unifying generative and discriminative information retrieval models. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, p. 515–524, New York, NY, USA, 2017. Association for Computing Machinery.

[43] Dong-Kyu Chae, Jin-Soo Kang, Sang-Wook Kim, and Jung-Tae Lee. Cfgan: A generic collaborative filtering framework based on generative adversarial networks. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, p. 137–146, New York, NY, USA, 2018. Association for Computing Machinery.

[44] Wenjie Wang, Yiyan Xu, Fuli Feng, Xinyu Lin, Xiangnan He, and Tat-Seng Chua. Diffusion recommender model. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, p. 832–841, New York, NY, USA, 2023. Association for Computing Machinery.

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

[46] Bin Li, Qiang Yang, and Xiangyang Xue. Transfer learning for collaborative filtering via a rating-matrix generative model. In *Proceedings of the 26th annual international conference on machine learning*, pp. 617–624, 2009.

[47] Weike Pan, Evan Xiang, Nathan Liu, and Qiang Yang. Transfer learning in collaborative filtering for sparsity reduction. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 24, pp. 230–235, 2010.

[48] Xianghui Zhu, Mengqun Jin, Hengyu Zhang, Chang Meng, Daoxin Zhang, and Xiu Li. Modeling domains as distributions with uncertainty for cross-domain recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, p. 2517–2521, New York, NY, USA, 2024. Association for Computing Machinery.

[49] Tong Man, Huawei Shen, Xiaolong Jin, and Xueqi Cheng. Cross-domain recommendation: An embedding and mapping approach. In *Ijcai*, Vol. 17, pp. 2464–2470, 2017.

[50] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1059–1068, 2018.

[51] Chao Li, Zhiyuan Liu, Mengmeng Wu, Yuchi Xu, Huan Zhao, Pipei Huang, Guoliang Kang, Qiwei Chen, Wei Li, and Dik Lun Lee. Multi-interest network with dynamic routing for recommendation at tmall. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pp. 2615–2623, 2019.

[52] Yukuo Cen, Jianwei Zhang, Xu Zou, Chang Zhou, Hongxia Yang, and Jie Tang. Controllable multi-interest framework for recommendation. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2942–2951, 2020.

[53] Luke Vilnis and Andrew McCallum. Word representations via gaussian embedding, 2015.

[54] Cédric Villani, et al. *Optimal transport: old and new*, Vol. 338. Springer, 2008.

[55] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, Vol. 11, No. 5-6, pp. 355–607, 2019.

[56] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, Vol. 26, , 2013.

[57] Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with sinkhorn divergences. In *International Conference on Artificial Intelligence and Statistics*, pp. 1608–1617. PMLR, 2018.

[58] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, Vol. 39, No. 9, pp. 1853–1865, 2016.

[59] Charlie Frogner, Chiyuan Zhang, Hossein Mobahi, Mauricio Araya, and Tomaso A Poggio. Learning with a wasserstein loss. *Advances in neural information processing systems*, Vol. 28, , 2015.

[60] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.

[61] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, Vol. 12, pp. 2825–2830, 2011.

Appendix

A   Additional Experimental Results

Publications

- 肖　子吟 and 鈴村 豊太郎 WebDB 夏のワークショップ 2025

Acknowledgment