Fan Liu
Zhenyang Li
Liqiang Nie

Multimodal Learning toward Recommendation

Springer

Multimodal Learning toward Recommendation

Fan Liu • Zhenyang Li • Liqiang Nie

Multimodal Learning toward Recommendation

Springer

Fan Liu
School of Computing
National University of Singapore
Singapore, Singapore

Zhenyang Li
Hong Kong Generative AI Research
and Development Center Limited
Hong Kong, Hong Kong

Liqiang Nie
Harbin Institute of Technology
Shenzhen, China

ISBN 978-3-031-83187-4 ISBN 978-3-031-83188-1 (eBook)
https://doi.org/10.1007/978-3-031-83188-1

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

With the explosive growth of online platforms, users are now faced with an overwhelming array of choices, making it increasingly difficult to quickly and accurately find the items they are looking for. To address this challenge and enhance the efficiency of information retrieval, recommender systems have become a cornerstone of modern web services and have been the subject of extensive research. The majority of existing recommender systems, particularly those based on collaborative filtering, predict users' preferences for unseen items by analyzing their historical interactions, achieving considerable success. However, in real-world scenarios, the user-item interactions are often sparse, which can complicate the prediction process and limit the effectiveness of traditional approaches. To overcome the sparsity issue, an increasing number of methods have been developed to leverage multimodal information related to users and items, such as images and reviews, to supplement user-item interactions and improve recommendation performance. These various modalities contain rich information about user preferences and item characteristics, which are highly beneficial for more accurate and fine-grained modeling of user preferences.

While multimodal information has the potential to significantly enhance recommender systems, effectively extracting and leveraging this information is not straightforward. The different modalities are often correlated rather than independent, with each providing a unique perspective on user preferences for items. As a result, multimodal learning is essential to fully harness the rich, recommendation-relevant information, rather than simply combining multiple unimodal data sources. While effective multimodal learning has been deeply explored in other multimodal tasks, its application in the recommendation domain is still in its early stages. Consequently, multimodal learning toward recommendation still faces the following critical challenges:

(a) Noise and Redundant Features. While multimodal information offers rich and valuable insights into users and items, it also contains a substantial amount of noise and redundant features. Extracting irrelevant multimodal data can hinder rather than enhance recommendation performance. Therefore, efficiently extracting recommendation-oriented features from multimodal data is essen-

tial for multimodal learning towards recommendation and poses a significant challenge.

(b) User Diverse Preference. Multimodal data often contains diverse characteristics of various items. Most existing methods rely on these multimodal features to jointly learn fixed user and item representations for recommendations, often overlooking the fact that users may focus on different aspects of items depending on their preferences. Since users exhibit diverse preferences across various items, accurately capturing these diverse preferences is crucial for accurate recommendations. As a result, a major challenge in multimodal learning is how to effectively explore and leverage the diverse characteristics of items embedded in multimodal features.

(c) User Modality Preference. For a given item, multiple modalities may provide relevant information, but users often have different preferences regarding which modalities they rely on. For instance, when purchasing clothing, some users may prefer to assess the style of clothing by viewing images, while others might rely more on the opinions expressed in reviews. As a result, the importance of different modalities in modeling user preferences can vary. Capturing these fine-grained preferences is essential for ensuring the robustness of recommendations. However, existing research often overlooks the user modality preference, significantly limiting the effectiveness of multimodal learning in recommendations.

(d) Dynamic Multimodal Fusion. Despite significant advances in multimodal fusion techniques, many existing approaches in recommendation systems still rely on static fusion strategies. These methods assume that the relationship between modalities remains constant, which may not hold true in real-world settings. In practice, the relevance and importance of different modalities can fluctuate over time due to factors such as changing user preferences, evolving contexts, and varying item characteristics. As a result, moving beyond static fusion models and exploring dynamic multimodal fusion approaches in recommendation is a significant challenge.

(e) Interpretability and Controllability. Recommender systems have long grappled with the challenge of interpretability, which is crucial for providing detailed explanations that enhance the trustworthiness of their recommendations. However, as is well known, most existing models—particularly those based on deep learning—are black-box models, which hampers efforts to improve interpretability. Moreover, the lack of interpretability significantly affects the controllability of recommender systems, making it difficult to generate targeted and desirable outcomes. Multimodal features inherently include attributes that represent specific, meaningful properties or characteristics of items. Therefore, leveraging multimodal information to improve the interpretability and controllability of recommendation systems is a critical challenge that multimodal learning urgently needs to address.

In this book, to tackle the aforementioned research challenges, we present some state-of-the-art multimodal learning theories towards recommendation. In particular, we first present a semantic-guided feature distillation method which employs a teacher-student framework to robustly extract effective recommendation-oriented

features from generic multimodal features. Next we introduce a novel multimodal attentive metric learning method to modal user diverse preferences for various items. This is achieved by a proposed attention neural network, which exploits the multimodal features of the target item to estimate a user's specific attention on each aspect of this item. After achieving efficient multimodal feature extraction and modeling of diverse user preferences, we then propose a disentangled multimodal representation learning recommendation model, which can capture users' fine-grained attention to different modalities on each factor in user preference modeling. Furthermore, we develop a meta-learning-based multimodal fusion framework to model the various relationships among multimodal information. Building on the success of disentangled representation learning, we further propose an attribute-driven disentangled representation learning method, which uses attributes to guide the disentanglement process to improve the interpretability and controllability of conventional recommendation methods. Finally, we conclude the book and figure out the future research directions in multimodal learning toward recommendation.

This book represents a preliminary research on multimodal learning towards recommendation, and we anticipate that the lectures in this series will dramatically influence future thought on these subjects. If in this book we have been able to dream further than others have, it is because we are standing on the shoulders of giants.

Singapore,
November 2024

Fan Liu
Zhenyang Li
Liqiang Nie

Acknowledgements

We gratefully acknowledge the numerous colleagues whose contributions have made this extensive book project both possible and enjoyable. In particular, we would like to express our sincere appreciation to the members of the N-CRiPT Lab at National University of Singapore and the iLearn Center at Shandong University, who have collaborated on various aspects of multimodal learning toward recommendation. Their efforts have provided valuable insights and contributed significantly to the discussions that informed the writing of this book.

Our first and foremost thanks undoubtedly go to colleagues who are also the contributors of some chapters of this book: Prof. Mohan Kankanhalli at National University of Singapore, Prof. Tat-Seng Chua at National University of Singapore, Prof. Zhiyong Cheng at Hefei University Of Technology, Dr. Yinwei Wei at Monash University, Dr. Wenjie Wang at National University of Singapore, Mr. HuiLin Chen at Hefei University Of Technology, and Miss. Changchang Sun at Illinois Institute of Technology. Thanks for their active participation in the technical discussion of this book and their constructive feedback and comments that have been significantly helpful in shaping this book. Particular thanks go to Dr. Han Liu at City University of Hong Kong, who engaged in in-depth discussions with us regarding the content of Chapter 5. Without his contributions, it would not have been possible to complete this book so smoothly.

Second, we are very grateful to the anonymous reviewer for taking the time to read the book despite their busy schedule.

Third, we sincerely extend our thanks to the Executive Editor, Mr. Ralf Gerstner, and the Product Editor, Ms. Ramya Prakash, whose support greatly contributed to making the book publishing process smooth and enjoyable.

Last, our thanks are reserved to our beloved families for their selfless consideration, endless love, and unconditional support.

Fan Liu, Zhenyang Li, and Liqiang Nie
November 2024

Contents

Authors' Biographies

Dr. Fan Liu is currently a Research Fellow with the School of Computing, National University of Singapore (NUS). He received his Ph.D degree from Shandong University in China. His research interests lie primarily in multimedia computing and information retrieval. His work has been published in a set of top forums, including ACM SIGIR, MM, WWW, TKDE, TOIS, TMM, and TCSVT. He is an area chair of ACM MM and a senior pc member of CIKM. Meanwhile, he has served as a technical program committee member for several top conferences, like ACM SIGIR, WWW, and ICML, and a reviewer for journals including TKDE, TOIS, TMM, and TCSVT.

Dr. Zhenyang Li is currently a Postdoc with the Hong Kong Generative Al Research and Development Center Limited. He received the Ph.D degree from Shandong University in China. His research interest lie primarily in recommendation and visual question answering. His work has been published in a set of top forums, including ACM MM, TIP, and TMM. He has served as the pc member for several top conferences, like ACM MM, SIGIR, ICLR, and the reviewer for journals including TKDE, TMM, TCSVT, ToMM.

Dr. Liqiang Nie, fellow of IAPR, is the director with the College of Information Science and Technology, Harbin Institute of Technology (Shenzhen campus). He received his Ph.D. degree from National University of Singapore. His research interests are multimedia content analysis and information retrieval. Dr. Nie has co-/authored more than 150 papers and 5 books, with 30,000 plus Google citations. He is an AE of IEEE T-KDE, IEEE T-MM, IEEE T-CSVT, ACM T-oMM, and Information Science. Meanwhile, he serves as the chair of ICMR 2025, ICME 2025, and ACM MM 2027. He is a member of ICME steering committee. He has received many awards over the past four years, like SIGMM rising star in 2020, MIT TR35 China 2020, SIGIR best student paper in 2021, IEEE AI's 10 to Watch in 2022, and ACM MM Best paper award in 2022.

Acronyms

BPR Bayesian Personalized Ranking
CE Cross Entropy
CF Collaborative Filtering
CNN Convolutional Neural Network
DNN Deep Neural Network
GCN Graph Convolution Network
KD Knowledge Distillation
LLM Large Language Model
MF Matrix Factorization
MLP Multi-Layer Perceptron
NDCG Normalized Discounted Cumulative Gain
RS Recommender System
SNN Shallow Neural Network

Chapter 1
Introduction

1.1 Background

The rapid advancement of Internet technology has fundamentally reshaped people's lifestyles, transforming how individuals access information, consume goods, and interact with others. The expansion of e-commerce platforms (e.g., Amazon [1], Taobao [2]), social networking platforms (e.g., Facebook [3], WeChat [4]), and streaming platforms (e.g., YouTube [5], TikTok [6]) has introduced a vast array of items (e.g., products, news), positioning these platforms as primary channels for consumption and information acquisition. This digital transformation has significantly contributed to the growth of the Internet economy, enabling users to seamlessly purchase clothing, read news, listen to music, watch movies, and book restaurants, hotels, or flights using mobile devices at any time.

The exponential growth in the number of items available on Internet platforms has provided users with extensive options, but it has also introduced challenges in allowing users to efficiently and accurately select desired items. These platforms often host tens of millions to billions of items. For instance, Amazon features over 353 million products, Taobao lists 800 million items daily, and Kuaishou processes 15 million new video uploads each day. Helping users discover their preferred items from these extensive catalogs is crucial for enhancing service quality, increasing user engagement, and maximizing platform revenue. To tackle these challenges, recommender systems have become core technologies on Internet platforms, providing personalized recommendations for products on e-commerce sites, content (e.g., news, movies, music) on streaming platforms, and friend/user suggestions on social networking platforms. The performance of these recommender systems is directly

[1] https://www.amazon.com/.

[2] https://www.taobao.com/.

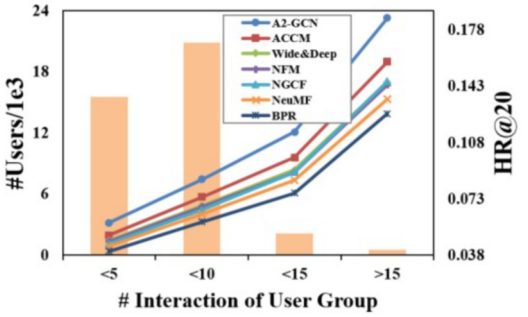
[3] https://www.facebook.com/.

[4] https://www.wechat.com/.

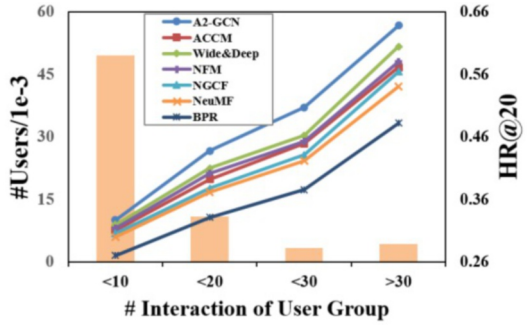
[5] https://www.youtube.com/.

[6] https://www.tiktok.com/.

F. Liu et al., *Multimodal Learning toward Recommendation*,
https://doi.org/10.1007/978-3-031-83188-1_1



(a) Amazon-Clothing



(b) Amazon-Kindle Store

Fig. 1.1: Data Sparsity Problem: The recommendation results of CF-based models across user groups with varying levels of interactions.

tied to platform revenue, drawing significant research interest from both academia and industry.

The primary goal of a recommender system is to effectively capture user preferences and predict items of interest from the vast array of available options, thereby delivering personalized recommendations to users. The most widely researched and applied approach in recommender systems is collaborative filtering (CF) [83, 9, 122, 54, 57, 21, 173, 56, 103, 102, 170, 108, 28, 105], which leverages historical interaction data between users and items to learn user preferences and item characteristics. CF-based algorithms identify and rank items that align with user preferences based on past interactions, which are generally categorized into explicit feedback (e.g., likes, purchases, ratings, and reviews) and implicit feedback (e.g., browsing, viewing, and clicking behaviors). Various CF-based recommendation algorithms have been proposed, including Probabilistic Matrix Factorization (PMF) and graph convolutional neural network approaches, which have significantly enhanced recommendation performance, improved user experience, and generated substantial economic benefits for Internet platforms.

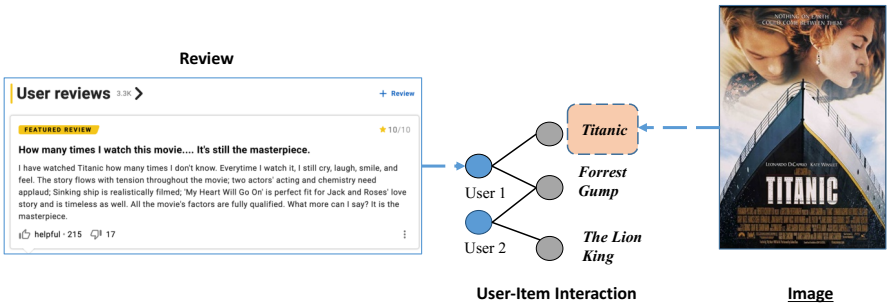


Fig. 1.2: Multimodal Data in Recommender Systems. Interactions, User Reviews, and Images for Movie Recommendations.

Despite their success, CF-based recommendation algorithms are heavily reliant on historical interaction data, making them effective only in scenarios with abundant user-item interactions. When faced with sparse data, commonly referred to as the data sparsity problem, the performance of these algorithms deteriorates significantly. As shown in Fig. 1.1, historical transaction data between users and items is often insufficient, with most items having only a limited number of interactions, resulting in a long-tail distribution problem. Based on two Amazon datasets, we compared the recommendation results of several CF-based recommendation models across user groups with varying levels of interactions. From the results, users interact with only a small fraction of available items, and as the number of interactions decreases, the system's performance declines sharply. Additionally, recommender systems encounter cold-start challenges when dealing with new users or items, making it difficult to accurately capture preferences and item characteristics based solely on historical data.

To address the data sparsity issue, researchers have increasingly focused on incorporating multimodal information related to users and items, supplementing the limited user-item interaction records [122, 54, 21, 211, 186, 185, 218]. As illustrated in Fig. 1.2, multimodal information, such as user-item interactions, user reviews and item images, contains rich contextual data that enhances the understanding of user preferences and item characteristics. User reviews provide textual insights into users' emotional attitudes toward items and describe specific item features, while item images visually depict attributes such as color, style, and design. Effectively extracting and utilizing the valuable information embedded in multimodal data can alleviate the data sparsity problem and enable recommender systems to more accurately capture user preferences. By integrating multimodal data, recommender systems are better equipped to mitigate the limitations posed by sparse interaction data, offering more accurate and robust recommendations.

In summary, multimodal learning has emerged as a pivotal research direction in the advancement of recommender systems. Enhancing the effectiveness of multimodal learning not only addresses the data sparsity issue inherent in traditional

recommendation algorithms but also leverages the vast amount of multimodal data available on Internet platforms. This approach holds significant potential for enhancing user experience and driving substantial economic growth.

1.2 Motivation

Multimodal information offers substantial potential for enhancing recommender systems by providing richer insights into both user preferences and item characteristics. However, simply combining multimodal data with traditional user-item interaction history is often inadequate for improving the performance of recommendation models. The various modalities are interrelated, rather than independent, each capturing distinct and complementary aspects of user preferences. Therefore, a robust multimodal learning approach is essential to fully exploit the rich, recommendation-oriented information contained within these modalities, rather than merely aggregating separate unimodal data sources.

Multimodal learning involves the integration and analysis of different types of modality data to derive more comprehensive and in-depth insights compared to those obtained from a single modality. It has been extensively explored in various tasks, such as Visual Question Answering and Image Captioning, leading to significant advancements. However, applying these multimodal learning methods to recommender systems is not straightforward. Effective multimodal learning in recommendations requires tailoring approaches to the unique characteristics of recommendation tasks, where user preferences are often implicit and highly dynamic. Currently, multimodal learning in recommender systems remains in its early stages, with significant challenges remaining in fully leveraging the diverse types of multimodal data for improved recommendation performance.

Given this context, this book aims to explore the specific requirements of multimodal learning for recommendation tasks from multiple perspectives. Furthermore, it seeks to propose novel methods that optimize the use of multimodal information in recommender systems, with the ultimate goal of improving the accuracy and personalization of recommendations.

1.3 Challenges

Multimodal learning in recommender systems demands careful design across several key dimensions, including feature extraction, modality feature fusion, and a thorough analysis of modality characteristics to meet the specific requirements of recommendation tasks. However, this process is fraught with challenges and is not easily achieved. Despite the significant progress made by existing multimodal learning methods in various domains, substantial difficulties remain in addressing the unique

needs of recommendation systems. These challenges can be broadly categorized into five main areas:

1.3.1 Noise and Redundant Features

One of the primary challenges in extracting multimodal features for recommendation systems lies in dealing with noise and redundant information inherent in multimodal data. While multimodal data provides rich insights into user preferences and item attributes, it also contains a considerable amount of noise and irrelevant content unrelated to the recommendation task. For example, user reviews may include off-topic discussions, and images may feature background elements unrelated to the item. If this irrelevant information is overly extracted, it may not only fail to enhance recommendation performance but could also impede the model's ability to effectively capture user preferences, ultimately leading to suboptimal recommendation outcomes.

Traditional feature extractors for different modalities struggle to effectively filter out noise and redundant information. Mainstream feature extractors for text, vision, and other modalities are typically based on pre-trained models from fields such as natural language processing and computer vision, which are not specifically designed or optimized for recommendation tasks. Consequently, the extracted features may not align well with the requirements of recommendation systems, limiting the model's capacity to deeply utilize these features and enhance performance.

Effectively filtering noise and redundant information, while ensuring efficient feature extraction, is therefore a crucial challenge for applying multimodal learning in recommendation systems. Removing noise and redundancy forms the foundation for fully leveraging multimodal information. By retaining only the information that is highly relevant to the recommendation task, models can more accurately capture user preferences, thereby utilizing multimodal data to enhance recommendation performance and user experience.

1.3.2 User Diverse Preference

The diversity of user preferences presents another significant challenge when utilizing multimodal information in recommendation systems. Traditional collaborative filtering algorithms often assume that user preferences are static and fixed, whereas in reality, user preferences are dynamic and vary across different types of items. Therefore, accurately capturing users' diverse preferences for various items is essential for achieving personalized recommendations.

Multimodal data offers a valuable opportunity to address this challenge by providing rich representations of user preferences. Users often express varied attitudes toward different attributes of an item in reviews, which provide important clues and

data for understanding their preferences. However, these diverse preferences are often embedded in different modalities (e.g., text, images, videos) and are not directly accessible or easy to utilize. This poses a challenge in multimodal learning—namely, how to effectively extract user preferences from each modality and integrate them into the recommendation model to leverage these preferences for more accurate recommendations.

1.3.3 User Modality Preference

In the process of multimodal learning, accurately capturing users' fine-grained preferences across different modalities is essential for effectively integrating information from each modality. This helps the recommendation system fully leverage multimodal data and enhances the robustness and accuracy of recommendation results, ultimately increasing user satisfaction.

However, modeling these fine-grained multimodal preferences is unique to recommendation contexts, and existing multimodal learning approaches have yet to provide effective solutions to address this issue. This limitation significantly restricts the efficacy of multimodal learning in recommendation systems. Future research should focus on developing models and algorithms capable of capturing users' detailed preferences across different modalities. By deeply analyzing user behavior within each modality, recommendation systems can better meet user needs and advance multimodal learning in the recommendation domain.

1.3.4 Dynamic Multimodal Fusion

It is well recognized that information from a single modality is often insufficient to comprehensively describe the characteristics of an item. For example, relying solely on textual descriptions may not fully capture the visual attributes of an item. Therefore, to achieve a more comprehensive understanding of item characteristics, multimodal recommender systems generally employ multimodal fusion techniques, leveraging the complementary nature of information across different modalities through integration.

Although multimodal fusion is crucial and numerous multimodal fusion methods have been proposed, existing approaches often employ a static multimodal fusion strategy to integrate different modality features. Specifically, for different items, the information from the same modality is assumed to play a similar role, and the relationships among different modality features are also considered invariant. However, this assumption is fragile. For instance, in case of clothing, images are often more reflective of the item's characteristics, whereas for books, textual modality are more effective in conveying the core content. As a result, existing static multimodal fusion methods often fail to account for the unique characteristics of different items

by applying the same fusion approach universally, making it difficult to achieve a more comprehensive item representation. Given these considerations, a promising direction for future research is to develop multimodal fusion strategies that account for the specific characteristics of different items, though this remains a significant challenge.

1.3.5 Interpretability and Controllability

Interpretability and controllability have long been essential objectives for building comprehensive recommendation systems. Interpretability refers to the ability of the recommendation model to provide clear and intuitive explanations for why specific content is suggested to the user. Such explanations enhance the transparency, persuasiveness, and effectiveness of recommender systems, fostering greater user trust and satisfaction. Controllability, meanwhile, refers to the system's flexibility in adjusting recommendations based on specific needs, constraints, or goals, allowing users or business stakeholders to influence recommendation strategies to achieve particular objectives or accommodate user preferences.

The incorporation of multimodal information presents new opportunities and challenges for enhancing the explainability and controllability of recommendations. On one hand, multimodal data contains rich information about user preferences and item attributes, such as text descriptions, images, audio, and videos, which provide valuable materials for explainability. For example, the system can visually justify recommendations by displaying product images or showing content that users have previously interacted with. On the other hand, multimodal data also provides more directly controllable attributes, allowing developers and users to adjust recommendation strategies for more refined control.

However, the complexity and high dimensionality of multimodal data pose challenges in achieving explainability and controllability. Multimodal data is inherently heterogeneous, with complex relationships and interactions between modalities, making it difficult to extract interpretable features. Furthermore, handling multimodal information requires more complex models, which may reduce model transparency and complicate understanding and control of recommendations.

To overcome these challenges, future research should focus on extracting interpretable features from multimodal data and designing models that can present recommendations in a clear and understandable manner, ultimately enhancing the controllability of the system. This would facilitate the development of multimodal recommendation systems capable of generating recommendations that users can understand, trust, and interact with meaningfully.

1.4 Our Solutions

In this book, to tackle the aforementioned research challenges, we present some state-of-the-art multimodal learning theories tailored to recommender systems. Our approach systematically enhances the capacity of recommendation systems to effectively leverage multimodal input information. To validate our proposed methods, we conducted comprehensive experiments on publicly available datasets, demonstrating significant improvements and robust performance across various scenarios. The results indicate that our techniques are well-suited to tackle the inherent complexities of multimodal data, thereby advancing the field of recommender systems.

1.4.1 Semantic-guided Feature Distillation

To address the challenges of noise and redundant information in multimodal data for recommend systems, we propose a semantic-guided feature distillation method aimed at efficiently extracting features pertinent to recommendation tasks. This method mitigates noise and redundancy by integrating knowledge distillation with semantic guidance, leading to more effective feature extraction.

Specifically, 1) to enhance the efficiency of feature extraction and minimize redundant information, we developed a teacher-student framework. Initially, a deep neural network with strong representation learning capabilities serves as the teacher network, effectively capturing essential features from each modality. Through knowledge distillation at both the feature-level and logits-level, the learned knowledge is then transferred to a student network composed of shallow neural networks, thereby improving the student's feature extraction capabilities for recommendation tasks. 2) Furthermore, to mitigate the effects of noise arising from insufficient consideration of semantic information in multimodal inputs, we incorporate item-specific semantic information into the feature extractor design. By introducing semantic guidance during the training process, the feature extractor is better equipped to focus on recommendation-relevant semantic information, effectively reducing the influence of irrelevant noise.

The synergy between these two modules enables efficient and targeted extraction of recommendation-relevant features from multimodal data. This approach can be seamlessly integrated into the feature extraction phase of any recommendation method, significantly enhancing the overall performance of recommendation systems.

1.4.2 Multimodal Attentive Metric Learning

To effectively capture users' fine-grained preferences for different items, we propose a multimodal attentive metric learning framework that dynamically leverages multi-

modal item information to determine user feature preferences during real-time interactions. This method employs attention mechanisms and metric learning to enhance the modelling ability and the geometric flexibility of user and item representations, respectively.

In particular, we design an attention neural network that utilizes both textual and visual information of items to estimate user preference weights for different item features. Each item is assigned a unique weight vector, enabling the modeling of diverse user preferences. Furthermore, to overcome the limitations of existing methods that do not satisfy the triangle inequality, we integrate attention weight vectors with metric learning to map user and item representations into a novel feature space. By utilizing Euclidean distance for nearest neighbor computation, we significantly improve the geometric flexibility.

The integration of these two components allows for a comprehensive and fine-grained capture of user preference diversity, ultimately enhancing the accuracy and effectiveness of recommendation systems.

1.4.3 Disentangled Multimodal Representation Learning

To effectively capture users' fine-grained preferences across different modalities, we propose a disentangled multimodal representation learning framework that quantifies the degree of user attention to various modalities of a factor. Specifically, this method first employs disentangled representation learning techniques to decompose the input from each modality into multiple independent and robust factor representations, establishing a solid foundation for capturing fine-grained multimodal preferences. Subsequently, based on these disentangled factor representations, we design a multimodal attention mechanism to accurately capture users' preferences for specific factors across different modalities. The resulting fine-grained multimodal preferences are integrated into recommendation systems, facilitating the efficient use of multimodal information and enhancing both the recommendations' robustness and effectiveness.

1.4.4 Dynamic Multimodal Fusion via Meta-Learning

To address the limitations of existing static multimodal fusion methods, which fail to adequately adapt to the unique characteristics of items, we propose a dynamic multimodal fusion approach that dynamically integrates multimodal information based on item-specific characteristics. In our approach, the multimodal fusion of each item is treated as an independent task, and we leverage meta-learning techniques along with items' meta information to customize the multimodal fusion module.

Specifically, our approach consists of two main components: a meta information extractor and a meta fusion learner. The meta information extractor, primarily im-

plemented using an multi-layer perceptron, extracts informative and concise meta information from multimodal features of the target item. The meta fusion learner then utilizes the extracted meta information to customize the parameters of the multimodal fusion neural network, thereby enabling dynamic multimodal fusion for different items.

1.4.5 Attribute-driven Disentangled Representation Learning

To improve the interpretability and controllability of recommendation results, we propose an attribute-driven disentangled representation learning framework designed to fully leverage the attribute information present in multimodal inputs. The decoupled representation learning process is divided into two main components: attribute-level decoupling and attribute-value-level structured representation learning. Based on the attributes of items, we divide the disentangled representation learning process into attribute-level disentangling and attribute value-level disentangling representation learning.

Specifically, attribute-level disentangled representation learning focuses on disentangling the representation of each modality according to the distinct attributes of items, resulting in independent factor representations with clear attribute-specific meanings. Following this, attribute value-level disentangled representation learning further refines these factor representations by considering the specific attribute values associated with each item, thereby enhancing the robustness and precision of the resulting representations.

By decoupling multimodal representations in this manner, we achieve representations that possess explicit attribute meanings, which can provide intuitive explanations for recommendation outcomes. Additionally, the independence among attribute-specific factors enhances the controllability of the recommender system, enabling more flexible adjustments of recommendation strategies to cater to specific user needs and preferences. This combination ultimately leads to a more transparent and adaptable recommendation process, improving both user satisfaction and system effectiveness.

1.5 Book Structure

In this book, we present an in-depth exploration of multimodal learning toward recommendation, along with a comprehensive survey of the most important research topics and state-of-the-art methods in this area. It is suitable for students, researchers, and practitioners who are interested in multimodal learning and recommender systems. It is worth emphasizing that the multimodal learning methods presented in this book are applicable to other retrieval or sorting related fields, like image retrieval, moment localization, and visual question answering.

The remainder of this book consists of five chapters. Chapter 2 presents the semantic-guided feature distillation method, which aims to address the challenges posed by noise and redundancy in multimodal data within recommender systems. This is a model-agnostic approach which offers improvements across a range of classical multimodal recommendation methods. Chapter 3 introduces the multimodal attentive metric learning framework designed to capture users' diverse preferences for different items at a fine-grained level. This method also resolves the prevalent issue in existing approaches related to the violation of the triangle inequality, thereby enhancing model robustness. In Chapter 4, we detail the disentangled multimodal representation learning approach, which aims to capture user preferences across different modalities with greater granularity. Chapter 5 presents a dynamic multimodal fusion approach that dynamically integrates multimodal information based on item-specific characteristics. Chapter 6 introduces an attribute-driven disentangled representation learning method, utilizing item attribute information to disentangle modality representations and assign explicit attribute semantics to each factor. This enhances both the interpretability and controllability of the recommendation results. Finally, Chapter 7 provides a comprehensive summary of the key contributions of the book and outlines promising future research directions for advancing multimodal learning in recommender systems.

Chapter 2
Semantic-Guided Feature Distillation for Multimodal Recommendation

2.1 Introduction

As introduced in Chapter 1, CF-based recommendation methods struggle to accurately capture user preferences accurately when the interaction data are sparse. This limitation arises because insufficient user-item interactions hinder the effective identification of similar users or items, thereby impacting recommendation performance. To address this issue, many recent approaches have begun extracting rich features related to user preferences and item characteristics from multimodal data [211, 63, 101, 182, 185, 99, 218, 171, 183]. Therefore, effectively extracting and integrating features from multimodal data serves as a foundation for a multimodal recommendation model.

Fig. 2.2 uses the visual feature extraction process to illustrate the general pipeline for feature extraction in the multimodal recommendation methods. Typically, pretrained models (such as VGG [145] for visual features and Bert [35] for textual features) are firstly adopted to extract generic features, which are then tailored for recommendation via a designed feature extractor (e.g., shallow/deep neural network). The multimodal features distilled from the feature extractor are subsequently used as input in the recommendation model to learn user and item representations. For example, to prune potential false-positive edges in the user-item interaction graph, Wei et al. [185] designed a method that scores the affinity between user preference and multimodal features of an item to measure the confidence of edge being true-positive interaction. It is worth mentioning that, in those methods, the feature extractor is often jointly learned with the user and item representations in the recommendation model in an end-to-end manner.

Despite the progress, many existing methods overlooked the importance of feature extraction in multimodal recommendation, resulting in interference from noise and redundant information during the extraction process. Specifically, these methods face two major challenges: first, regardless of whether shallow or deep neural networks are employed for feature extraction, it remains challenging to meet the specific requirements of recommendation scenarios. Due to the complexity and high

F. Liu et al., *Multimodal Learning toward Recommendation*,
https://doi.org/10.1007/978-3-031-83188-1_2

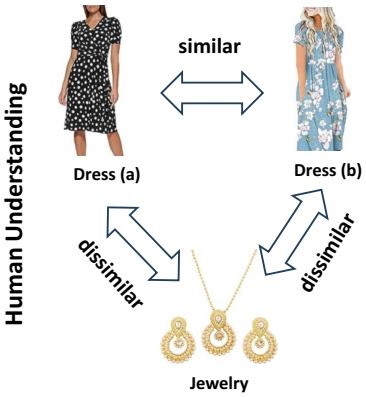


Fig. 2.1: Relationship between items with considering semantic information.

dimensionality of multi-modal features, a compact feature extractor adopting shallow neural networks (SNNs) can hardly extract effective information from the generic features of each modality. Therefore, most recent multimodal recommendation models employ deep neural networks as the extractor [211, 63, 101, 182, 185], leveraging the powerful representation learning capability of DNNs. However, the problem is that deep models need sufficient training data for good performance. Unfortunately, recommendation models often face serious data sparsity problem. As a result, feature extractors with deep models cannot be well trained, which not only renders them unhelpful but can also negatively impact the final recommendation performance. The second challenge is that semantic information is often ignored during the feature extraction process. In fact, existing recommendation models extract modality features only using collaborative information [101, 185, 218]. Yet, semantic information contains rich semantic and concise knowledge, which can well guide the feature extraction process, helping the feature extractor focus on relevant information while avoiding interference from noise and redundant data. As illustrated in Fig. 2.1, category information is a more abstract and general form of semantic information, and it can help recommendation models identify relationships or similarities between items.

Inspired by the above considerations, in this chapter, we introduce a novel model-agnostic approach, termed **S**emantic-**G**uided **F**eature **D**istillation (SGFD for short), which can robustly extract effective recommendation-oriented features from generic multimodal features using the teacher-student framework. Specifically, we propose a teacher model composed of feature extractors with deep neural networks for different modalities. These feature extractors are designed to extract features considering both the semantic information of items and the complementary information of multiple modalities. As the teacher model is separately trained with recommendation model, it will not suffer the data sparsity problem in recommendation. Meanwhile, we use the feature extractors with shallow neural networks as the student model to replace the original feature extractors in recommendation methods. In addition, we

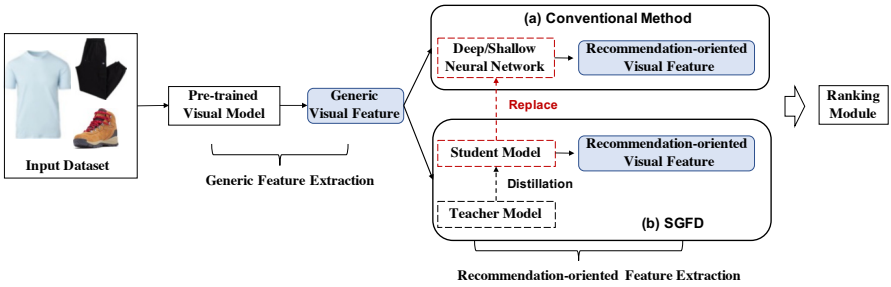


Fig. 2.2: An illustration of the feature extraction pipeline in multimodal recommendation models. (a) is the conventional feature extraction method and (b) is our proposed SGFD.

introduce a feature-based distillation loss and a response-based distillation loss to transfer knowledge from the teacher model to the student model. To evaluate the efficacy of our approach, we conducted comprehensive experiments on three real-world datasets. The results demonstrate that our method can significantly outperform existing multimodal recommendation models. We have released the code and relevant parameter settings to facilitate repeatability as well as further research [1].

In summary, the main contributions of this work are as follows:

- We emphasize the constraints of existing recommendation methods in effectively extracting multimodal features from data that contain noise and redundant information. To overcome these limitations, we propose a model-agnostic approach SGFD that can extract effective recommendation-oriented features from generic modality features using the teacher-student framework.
- To leverage the semantic information during the feature extraction processes, we design feature extractors that consider both the semantic information of items and the complementary information of multiple modalities.
- We employ the response-based and feature-based distillation loss to facilitate knowledge transfer from the teacher to the student model.
- We have conducted extensive experiments on three real-world datasets, demonstrating the superiority of our method.

The rest of this chapter is structured as follows: Section 2.2 provides a thorough review of the related work. Section 2.3 introduces the preliminaries relevant to our proposed approach. Next, we introduce our proposed approach and then provide details on its two main components: multimodal feature extraction and semantic-guided feature distillation in Section 2.4. We then present the results of our experiment in Section 2.5. Finally, we conclude this chapter in Section 2.6.

[1] https://github.com/HuilinChenJN/SGFD.

2.2 Related Work

In this section, we briefly introduce the recent advances of model-based CF models, multimedia recommendation and knowledge distillation, which are closely relevant to our work.

2.2.1 Model-based Collaborative Filtering

In model-based CF models, users and items are represented as dense vectors (i.e., embeddings) in the same latent space. Based on the learned vectors, an interaction function is used to predict the preference of a user to an item. Take the Matrix Factorization (MF) for example, the user and item embeddings are learned by minimizing the error of re-constructing the user-item interaction matrix, and the dot product is used as the interaction function for prediction. This simple idea has achieved great success, and many variants of MF has been developed [136, 134, 57, 169, 172, 213]. Inspired by this simple idea, many variants of MF has been developed, such as BPR [134], PMF [136], WRMF [66], etc. The advent of deep learning has accelerated the development of model-based CF techniques. Due to the powerful capability of deep learning, it has been widely applied in recommendation to learn better user and item embeddings [193, 29, 100] or model more complicate interactions between users and items [57, 90]. For example, NeuMF [57] models the complex interactions between users and items using nonlinear neural networks as the interaction function. More recently, Graph Convolution Network (GCN) techniques have also been applied in recommendation and set a new standard for recommendation [10, 173, 103, 102, 194, 195, 105]. The advantage of GCN-based recommendation models is attributed to its capability of explicitly modeling high-order proximity between users and items. For example, GC-MC [10] employs one convolution layer to exploit the direct connections between users and items. As a typical method, NGCF [173] exploits high-order proximity by propagating embeddings on the user-item interaction graph to directly facilitate the user and item embedding learning. To alleviate the over-smoothing problem in GCN-based recommendation models, IMP-GCN [102] performs high-order graph convolution inside sub-graphs of users with similar interests.

All of the aforementioned methods learn the representations of users and items merely relying on the interaction data. And their performance suffers when the interactions are sparse. Besides the interaction data, the rich side information, such as reviews and images, provide valuable information of user preference and item characteristics, which has been widely used to alleviate the data sparsity problem in recommendation [121, 24, 155, 21].

2.2.2 Feature Extraction in Multimodal Recommendation

Although multimodal data contains rich information about user preferences and item characteristics, only by effectively extracting these meaningful features can recommendation systems make full use of this information. Typically, in multimodal recommendation, the initial step for processing the raw modality inputs (*e.g.*, images and text) it to obtain dense feature representations using various modality encoders. For visual inputs, early approaches primarily employed CNN-based pre-trained models as visual encoders. These visual encoders extract and compress visual features from raw pixel data through convolution and pooling operations. For example, MMGCN [186] adopts ResNet [53], whereas VECF [23] uses VGG [145] as the visual encoder. Recently, Transformer-based visual pre-trained models have demonstrated superior performance [81, 203, 50], leading to their adoption as visual encoders in multimodal recommender systems. For instance, ODMT [70] employs ViT [81] as its visual encoder. For textual modality, such as user reviews and item description, multimodal recommendation methods typically utilize feature extraction techniques such as Word2Vec [123], RNN [62] and BERT [35]. More recently, with the significant success of LLMs in natural language processing [12], researchers have begun to explore the use of LLMs for extracting textual features of items [216].

After extracting generic visual and textual features from raw images and text, these features are further processed by well-designed feature extractors to derive recommendation-specific feature vectors. Generally, multimodal recommender systems employ either SNNs or DNNs as feature extractors [101, 211, 63]. However, SNNs lack sufficient capacity for effective feature extraction, while DNNs, despite their powerful representation learning capability, face the challenge of data sparsity in recommendations. Both struggle to meet the requirements of recommendation, especially in the presence of noise and redundant data. To overcome this limitation, this study leverages knowledge distillation to simultaneously exploit the advantages of both SNNs and DNNs, thereby enhancing the quality of feature extraction.

2.2.3 Knowledge Distillation

Knowledge Distillation (KD) is composed of a "compact" model (i.e., student model) and a previously trained "large" model (i.e., teacher model) [61, 82, 2, 127]. It is widely applied in various fields, especially in image classification tasks [220, 127]. The core of KD is that a previously trained large teacher model transfers knowledge to a small student model, so that the student model performs comparably to the teacher model. Moreover, the student model also has lower inference latency due to its small size and compact network architecture. Feature Distillation (FD) is a primary method in KD, which distill features in the teacher network through a transformation [135, 2, 158]. For example, TOFD [208] is proposed to compress and accelerate the neural networks by transferring information in convolutional layers from the teacher model to the student one.

Recently, inspired by the success of KD in various fields, it has been introduced into the Recommendation System (RS) to enhance the capability of modeling users' preferences or reduce inference latency while maintaining performance [87, 156, 74], such as Ranking Distillation (RD), Collaborative Distillation (CD). Ranking Distillation (RD) learns to rank documents/items from both the training data and the supervision of a larger teacher model [156, 74]. For example, Kang et al. [74] proposed a method that enables the student model to learn from the latent knowledge encoded in the teacher model and from the teacher's predictions results. Collaborative Distillation (CD) first samples unobserved items from the teacher's recommendation list according to their ranking, then trains the student to mimic the teacher's prediction score (e.g., relevance probability) on the sampled items [87].

In this chapter, we propose a teacher model which extracts multimodal features from the generic feature, guided by semantic information. Meanwhile, we employ a student model with shallow neural networks to serve as feature extractors for various modalities in recommendation model.

2.3 Preliminaries

2.3.1 Task Formulation

In this chapter, we focus on feature extraction in multimodal recommendation. To alleviate the data sparsity problem in recommendation, multimodal features (e.g., user review and item image [2]) are leveraged to learn better representations of users and items. In multimodal recommendation, the generic review and image features are extracted from multimodal data using pre-trained models. The main objective of our work is to derive effective recommendation-oriented features from these generic multimodal features to improve recommendation accuracy. In other words, our task is to effectively utilize the information contained in the reviews and images of items to better capture users' preferences and make more accurate recommendations.

2.3.2 Generic Feature Extraction

In our implementation, we use the VGG16 model [145] to extract generic visual features from item images. This model consists of 13 convolutional layers and three fully connected layers. It is trained by using 1.2 million ImageNet (ILSVRC2010) images. We utilize the output from the second fully connected layer, a 4096-dimensional feature vector, as the generic visual feature $\mathbf{F}_m$:

[2] Note that only the multimodal data of items is used in this work. Besides, each item is associated with one image.

$$\mathbf{F}_m = \text{VGG16}(\mathbf{I}), \tag{2.1}$$

where $\mathbf{I}$ represents the input item image.

Additionally, we employ BERT [35] to extract generic textual features. This model is a groundbreaking pre-trained language model developed by Google that uses bidirectional attention to understand the context of words in a sentence. By processing text in both directions, it captures richer contextual information, making it highly effective for various NLP tasks such as question answering, sentiment analysis, and named entity recognition. BERT's pre-training on large text corpora and fine-tuning for specific tasks has set new performance benchmarks in the field, significantly advancing natural language understanding. In our work, we input all the users' reviews for an item to BERT to obtain the generic textual feature $\mathbf{F}_e$ for this item:

$$\mathbf{F}_e = \text{BERT}(\mathbf{R}), \tag{2.2}$$

where $\mathbf{R}$ represents the aggregated reviews.

2.3.3 Recommendation-oriented Feature Extraction

After obtaining the generic visual and textual features $\mathbf{F}_m$ and $\mathbf{F}_e$, the feature extractors in the multimodal recommendation model are then used to tailor these features for recommendation. Take visual feature as an example, previous studies [185, 101, 218] feed the generic visual feature $\mathbf{F}_m$ into a Multi-Layer Perceptrons (MLP) to extract the recommendation-oriented visual feature $\mathbf{f}_m$. Specifically, it is defined as:

$$\mathbf{f}_m = \text{MLP}(\mathbf{F}_m), \tag{2.3}$$

It should be noted that the adopted $\text{MLP}(\cdot)$ are jointly optimized with the recommendation model. Thus, the training of feature extractors will also suffer from the data sparsity problem in recommendation. The recommendation-oriented textual feature $\mathbf{f}_e$ can be obtained similarly:

$$\mathbf{f}_e = \text{MLP}(\mathbf{F}_e), \tag{2.4}$$

In multimodal recommendation models, the textual feature $\mathbf{F}_e$ and visual feature $\mathbf{F}_m$ will be fed into two separate MLPs to extract the effective recommendation-oriented features $\mathbf{f}_e$ and $\mathbf{f}_m$, respectively. These extracted features $\mathbf{f}_e$ and $\mathbf{f}_m$ are subsequently used to improve recommendation accuracy.

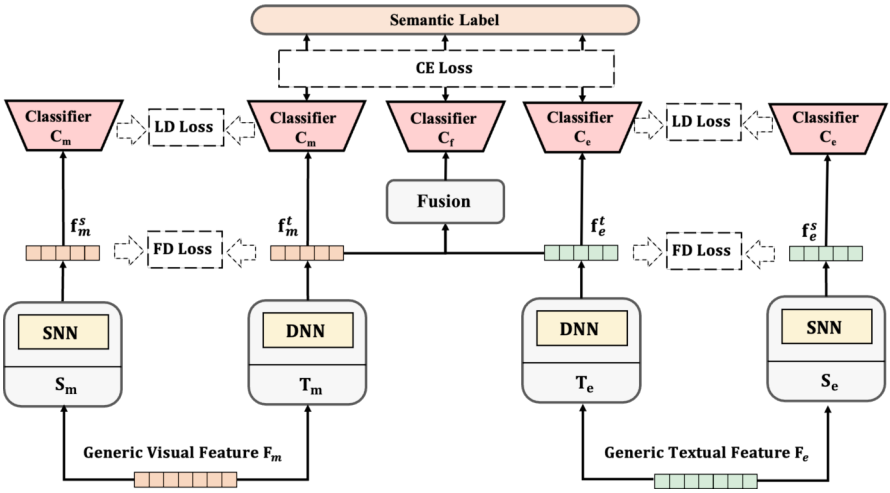


Fig. 2.3: An overview of the proposed Semantic-Guided Feature Distillation (SGFD) method. Taking visual feature extraction as an example, the visual feature extractor in the teacher model extracts features from generic visual features considering both the semantic and complementary information. The Logits-based Distillation (LD) loss and the Feature-based Distillation (FD) loss are employed to transfer the knowledge to the student model.

2.4 Semantic-guided Feature Distillation

2.4.1 Overview

We propose a model-agnostic approach, termed semantic-guided feature distillation, to extract effective recommendation-oriented features from generic multimodal features for recommendation. As shown in Fig. 2.3, the proposed method consists of a teacher model and a student model. In the teacher model, the DNN-based feature extractors (T_e and T_m) for textual and visual features are designed to extract effective features considering the semantic and complementary information. In our cases, the feature extractor in the student model replaces the original extractor in multimodal recommendation models. In other words, the SNN-based feature extractors (S_e and S_m) in the student model are part of multimodal recommendation models. Therefore, it can obtain the semantic information from teacher extractors by the adopted knowledge distillation technique and benefit from the user behaviors. Next, we detail these two models and the knowledge distillation technique.

2.4.2 Teacher Model

Since semantic information is often overlooked while extracting features from multimodal data for recommendation, we propose a teacher model that extracts effective features considering the semantic information of items. Specifically, we leverage semantic labels (e.g., category labels) as supervised information to classify the extracted features using classifiers. Our proposed method employs three distinct classifiers $\mathrm{C}_e(\cdot)$, $\mathrm{C}_m(\cdot)$ and $\mathrm{C}_f(\cdot)$ for textual, visual, and fused features. Moreover, considering that the generic modality feature is highly complex and high dimensional, we utilize deep neural networks as feature extractors for visual and textual features, denoted as $\mathrm{T}_m(\cdot)$ and $\mathrm{T}_e(\cdot)$, respectively.

Extracting Semantic-aware Features

Take the generic visual feature $\mathbf{F}_m \in \mathcal{R}^{D_m}$ as an example, we first feed $\mathbf{F}_m$ into the feature extractor to obtain valuable features $\mathbf{f}_m^t \in \mathcal{R}^{d_m}$:

$$\mathbf{f}_m^t = \mathrm{T}_m(\mathbf{F}_m), \tag{2.5}$$

where T_m is the feature extractor that consists of a deep neural network with k layers [3]. Similarly, we obtain the textual feature $\mathbf{f}_e^t$ from the generic textual feature using the textual feature extractor T_e. The extracted visual and textual features are then separately fed into the classifier $\mathrm{C}_m(\cdot)$ and $\mathrm{C}_e(\cdot)$. Take the visual feature $\mathbf{f}_m^t$ as an example, it is formulated as follows:

$$\mathbf{P}_m^t = \mathrm{C}_m(\mathbf{f}_m^t; \tau) = \mathrm{softmax}((\mathbf{W}_m \mathbf{f}_m^t + \mathbf{b}_m)/\tau), \tag{2.6}$$

where $\mathbf{P}_m^t$ represents the probability distribution output for the visual feature. $\mathbf{W}_m$ and $\mathbf{b}_m$ are the trainable weight matrices and bias vectors, respectively. It should be noted that the classifier parameters for each modality are trained independently. Here, τ refers to a temperature hyperparameter that controls the smoothness of the distribution. Similarly, the probability distribution output for the textual feature $\mathbf{P}_e^t$ can be obtained.

The feature extractors are optimized using cross-entropy (CE) loss. The combined CE loss for textual and visual features is formulated as follows:

$$\mathcal{L}_{TCE} = (\mathrm{CE}(\mathbf{P}_m^t, \mathbf{y}) + \mathrm{CE}(\mathbf{P}_e^t, \mathbf{y}))/2, \tag{2.7}$$

where $\mathrm{CE}(\hat{\mathbf{y}}, \mathbf{y}) = -\sum_{i=1}^{C} y_i \cdot log(\hat{y}_i)$ denotes the cross-entropy loss. C is the number of classes, $\mathbf{y}$ is the ground-truth label, and $\hat{\mathbf{y}}$ is the predicted probability distribution.

Exploiting Complementary Information

To enhance the ability of teacher extractors to extract effective features of each

[3] k is calculated by $k = \log_\delta \frac{D_m}{d_m}$, where δ is a hyperparameter that controls the number of layers based on the input dimension of each modality.

modality, we first leverage complementary information between multiple modalities by fusing textual and visual features. As shown in Fig. 2.3, taking the output of each modality extractor as input, we fuse the vectors as follows:

$$\mathbf{f}_u = \sigma(\mathbf{W}_u[\mathbf{f}_m^t; \mathbf{f}_e^t]) + \mathbf{b}_u), \tag{2.8}$$

where $\mathbf{W}_u$ and $\mathbf{b}_u$ denote the trainable weight matrix and bias vector. $\sigma(\cdot)$ is the activation function and ReLU is adopted because of its biologically plausible and non-saturated property [190]. Then, the fused feature $\mathbf{f}_u$ is fed into the classifier $\mathrm{C}_f(\cdot)$, and the CE loss function of the fused feature is defined as follows:

$$\mathbf{P}_u = \mathrm{C}_f(\mathbf{f}_u; \tau), \tag{2.9}$$

$$\mathcal{L}_{FCE} = \mathrm{CE}(\mathbf{P}_u, \mathbf{y}). \tag{2.10}$$

The weight matrix and the bias vector of $\mathrm{C}_f(\cdot)$ are trainable and specifically trained with fused features to capture the complementary information between multiple modalities. The temperature hyperparameter τ is adopted similarly as they are in Eq. 2.7.

2.4.3 Student Model

To reduce the negative effect of the data sparsity problem on feature extraction in recommendation, we utilize shallow neural networks (SNNs) as feature extractors for generic multimodal features. The feature extractors for visual and textual features in the student model are denoted as $\mathrm{S}_m(\cdot)$ and $\mathrm{S}_e(\cdot)$, respectively. For the generic visual feature $\mathbf{F}_m$, the recommendation-oriented visual feature $\mathbf{f}_m^s$ is obtained by:

$$\mathbf{f}_m^s = \mathrm{S}_m(\mathbf{F}_m). \tag{2.11}$$

The textual feature $\boldsymbol{f}_{\boldsymbol{e}}^{\boldsymbol{s}}$ can be obtained in the same manner. Then, the visual and textual features $\mathbf{f}_m^s$, $\mathbf{f}_e^s$ are fed into the classifier $\mathrm{C}_m(\cdot)$ and $\mathrm{C}_e(\cdot)$ respectively. Taking the visual feature as an example, it is formulated as:

$$\mathbf{P}_m^s = \mathrm{C}_m(\mathbf{f}_m^s; \tau). \tag{2.12}$$

By following this way, we can obtain $\boldsymbol{P}_e^s$ for the textual feature. Note that the classifiers adopted are the same as Eq.2.6.

2.4.4 Knowledge Distillation

The proposed teacher model, which utilizes deep neural networks, is capable of extracting semantic-aware features from the generic features of each modality. On

the other hand, the student model, employing SNNs, can alleviate the negative effect of the data sparsity problem on feature extraction in recommendation. To leverage the strengths of both the teacher and student models, we employ the knowledge distillation technique to transfer the response- and feature-based knowledge encoded in the teacher model to the student model. Next, we detail the two distillation losses used for transferring this knowledge.

Response-Based Knowledge

Response-based knowledge refers to the neural response (logits) generated by the final output layer of the teacher model. In our method, we consider the output of the classifiers as response-based knowledge. Hence, we employ the Logits-based Distillation (LD) loss to enable the feature extractors in the student model to mimic the predicted label distribution of the feature extractors in the teacher model. The distillation loss between the feature extractors of visual and textual features for both the teacher and student models can be described as follows:

$$\mathcal{L}_{LD} = ||\mathbf{P}_m^t - \mathbf{P}_m^s||_2^2 + ||\mathbf{P}_e^t - \mathbf{P}_e^s||_2^2. \tag{2.13}$$

Note that we do not force the student model to match the prediction distribution of the teacher model; instead, we expect the student model to gain semantic information via knowledge transfer. Thus, following the previous work [61], we use soft targets to embody the informative knowledge derived from the teacher model.

Feature-Based Knowledge

Given the capacity of deep neural networks for feature extraction, the output from intermediate layers can serve as knowledge to guide the training of the student model. In our method, we employ the Feature-based Distillation (FD) loss to transfer the visual and textual feature-based knowledge from the teacher model to the student model. The distillation loss can be formulated as follows:

$$\mathcal{L}_{FD} = ||\mathbf{f}_m^t - \mathbf{f}_m^s||_2^2 + ||\mathbf{f}_e^t - \mathbf{f}_e^s||_2^2. \tag{2.14}$$

2.4.5 Optimization

In this work, we target the top-n recommendation and equip our proposed SGFD with three multimodal recommendation models (MAML [101], GRCN [185], BM3 [218]). However, these three recommendation models have different objective functions. Therefore, we refer to the loss of objective functions of these three recommendation models as $\mathcal{L}_{REC}$.

For our proposed SGFD, the objective function consists of the Cross-Entropy Loss in the teacher model and the Knowledge Distillation Loss. Specifically, the total Cross-Entropy Loss is formulated as:

$$\mathcal{L}_{CE} = \mathcal{L}_{TCE} + \mathcal{L}_{FCE}, \tag{2.15}$$

The Knowledge Distillation Loss utilized for knowledge transfer is formulated as:

$$\mathcal{L}_{KD} = \mathcal{L}_{LD} + \mathcal{L}_{FD}, \tag{2.16}$$

The student model of the proposed SGFD is jointly trained with the multimodal recommendation model in an end-to-end manner. Overall, the final loss function of the recommendation model equipped with SGFD is,

$$\mathcal{L}_{SGFD} = \mathcal{L}_{REC} + \lambda_1 \mathcal{L}_{KD} + \lambda_2 \mathcal{L}_{CE}, \tag{2.17}$$

where λ_1 and λ_2 are used to adjust the weight of the knowledge distillation loss and cross-entropy loss.

2.5 Experiments

To validate the effectiveness of our method, we conducted extensive experiments on four public datasets to answer the following research questions:

RQ1: How does feature distillation affect the performance of multimodal recommendation models?

RQ2: Does our proposed SGFD improve the performance of state-of-the-art multimodal recommendation models on the top-n recommendation task?

RQ3: How do various components in SGFD affect the effectiveness of feature distillation?

RQ4: Can our method effectively transfer semantic knowledge for multi-modal features?

Next, we first introduce the experimental setup and then report and analyze the experimental results.

2.5.1 Experimental Setup

Datasets

In experiments, we utilize the Amazon review dataset [4] for experimental evaluation as in previous studies [121]. The Amazon Review Dataset is a large collection of product reviews from Amazon, which includes user-generated feedback on various products across different categories. It is widely used in data science, machine learning, and natural language processing (NLP) tasks, particularly for sentiment analysis, recommendation systems, and text mining. This public dataset contains

[4] http://jmcauley.ucsd.edu/data/amazon.

Table 2.1: Basic statistics of the experimental datasets.

Dataset	#user	#item	#label	#interactions	sparsity
Office	4,874	7,279	54	52,957	99.85%
Clothing	18,209	17,318	26	150,889	99.98%
Toys Games	18,748	11,672	19	161,653	99.97%
Sports	21,400	36,224	18	982,618	99.97%

user interactions (review, rating, helpfulness votes, etc.) on items and the item metadata (descriptions, attributes, images, etc.) on 24 product categories. To verify the effectiveness of our proposed method, we choose the three per-category datasets (*Office*, *Clothing*, *Toys&Games* and *Sports*) for performance evaluation under different sizes and sparsity levels. In these datasets, we treat each rating as a positive user interaction. In contrast, we treat all the items the user did not interact with as negative. And then, we pre-processed the dataset with a 5-core setting on both items and users in our settings. All user reviews and item images are used as item-side information. Table 2.1 shows the basic statistics of the four datasets.

In this chapter, we focus on recommending a set of top-N ranked items that will appeal to the user. Specifically, we randomly sampled 80% of the interactions of each user to create the training set and retained 20% of the user interactions for testing.

Baselines

To demonstrate the effectiveness of our proposed SGFD, we first equipped it on three existing multimodal recommendation models: GRCN [185] and BM3 [218]. Then, we compared the performance of newly built models (SGFD$_{GRCN}$, SGFD$_{BM3}$) with the following baselines, including general CF recommendation models (e.g., NeuMF [57], CML [63], NGCF [173] and DGCF [174]) and multimodal recommendation models (e.g., GRCN, BM3).

- **NeuMF [57]**: This neural collaborative filtering method employs multiple hidden layers above the element-wise and concatenation of user and item embeddings to capture their non-linear feature interactions. Especially, it uses two-layered plain architecture, where the dimension of each hidden layer keeps the same.
- **CML [63]**: This model learns a joint metric space to encode users' preferences and the user-user and item-item similarity for top-n recommendation.
- **NGCF[173]**: This GCN-based recommendation model explicitly encodes the collaborative signal of high-order neighbors by performing embedding propagation in the user-item bipartite graph.
- **DGCF [174]**: This GCN-based method refines the interaction graphs considering the users' diverse intents. Meanwhile, it encourages independence of different intents by adopting disentangled representation technique.

- **GRCN [185]**: This GCN-based multimedia recommendation model adjusts the structure of interaction graph adaptively based on the status of model training for discovering and pruning potential false-positive edges.
- **BM3 [218]**: This self-supervised framework uses a latent representation dropout mechanism to generate the target view of a user or an item. BM3 is trained without negative samples by jointly optimizing three multi-modal objectives.

During the evaluation, we put great effort into tuning the hyperparameters on the released code of all baselines and the best performance are reported.

Evaluation Metrics

Two widely-used evaluation metrics are adopted for top-n recommendation: *Recall* and *Normalized Discounted Cumulative Gain* (NDCG) [55]. *Recall* measures the proportion of relevant items successfully retrieved by the recommendation algorithm. It is defined as:

$$\text{Recall} = \frac{|\text{Rel}(u) \cap \text{Rec}(u,n)|}{|\text{Rel}(u)|}, \tag{2.18}$$

where $\text{Rel}(u)$ is the set of relevant items for user u, and $\text{Rec}(u,n)$ is the set of top-n recommended items for user u. The *Ideal Discounted Cumulative Gain* (IDCG) is defined as:

$$\text{IDCG@}n = \sum_{i=1}^{|\text{Rel}(u)|} \frac{2^{\text{rel}_i} - 1}{\log_2(i+1)}, \tag{2.19}$$

Finally, the *Normalized Discounted Cumulative Gain* (NDCG) is given by:

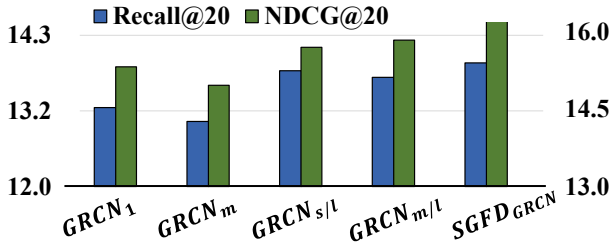
$$\text{NDCG@}n = \frac{\text{DCG@}n}{\text{IDCG@}n}. \tag{2.20}$$

Recommendation accuracy is calculated for each metric based on the top 20 results. Note that the reported results are the average values across all testing users.

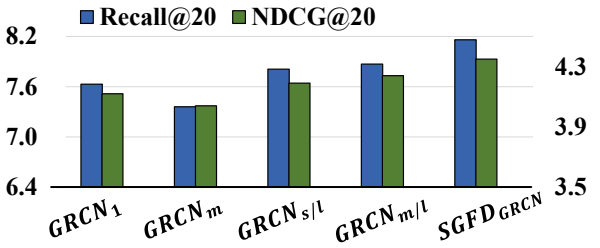
Parameter Settings

The PyTorch framework [5] is adopted to implement the proposed model. In our experiments, all hyperparameters are carefully tuned. For all datasets, the embedding size of the users and items is set to 128. The mini-batch size is fixed to 1024. The learning rate for the optimizer is searched from $\{10^{-5}, 10^{-4}, \cdots, 10^{+1}\}$, and the model weight decay is searched in the range $\{10^{-5}, 10^{-4}, \cdots, 10^{-1}\}$. For knowledge transfer, the temperature τ is tuned from$\{5, 10, 20, 50, 100\}$, while for classification, it is set to 1. The layer number of the neural networks in the teacher model δ is tuned from $\{2, 4, 6, 8\}$. Moreover, the weights λ_1 and λ_2 are searched in the range $\{10^{-3}, 10^{-2}, \cdots, 10^{+1}\}$. Besides, the early stopping strategy is adopted as previous

[5] https://pytorch.org.



(a) Office



(b) Clothing

Fig. 2.4: Performance comparison of GRCN equipped with different feature extraction methods. Notice that the values are reported by percentage with '%' omitted.

studies [173, 102]. Specifically, the training process will stop if recall@20 does not increase for 20 successive epochs.

2.5.2 Study of Multimodal Feature Extraction in Recommendation (RQ1)

In this section, we primarily investigate the influence of different feature extractors for multimodal features on recommendation performance. We compare the performance of four feature extraction methods with our proposed SGFD based on GRCN, and the results are reported in Fig. 2.4. Our analyses are based on performance comparisons to the following methods:

- **GRCN**$_1$ **[185]**: It is the vanilla GRCN method that adopts the shallow neural network as the feature extractor.
- **GRCN**$_m$: The vanilla GRCN method adopts the deep neural network as the feature extractor.
- **GRCN**$_{s/l}$: Given the vanilla GRCN method, we employ the teacher model of SGFD with shallow neural networks as the feature extractor in GRCN.
- **GRCN**$_{m/l}$: Given the vanilla GRCN method, we employ the teacher model of SGFD with the deep neural networks as the feature extractor in GRCN.

From the results, we can see that GRCN_1 yields better performance than GRCN_m on *Office* and *Clothing*. Furthermore, the performance degradation of GRCN_m in the *Office* is less compared to *Clothing*. This can be attributed to the fact that the feature extractor in most multimodal recommendation models is jointly optimized with recommendation models. Consequently, the feature extractor will suffer the data sparsity problem when the user-item interactions are sparsity. As *Office* is denser, the data sparsity problem less impacts the feature extraction model. The variant $\text{GRCN}_{s/l}$ outperforms GRCN_1 and GRCN_m across all the datasets, which indicates that the semantic information has positive advancement in extracting valuable features for recommendation. More importantly, by adopting multiple neural networks and introducing the complementary information of multimodal features, the variant $\text{GRCN}_{m/l}$ has an explainable improvement on *Clothing*. This demonstrates that the complementary information is facilitated to reduce redundant information derived from the feature extractor in each modality. But, it still underperforms $\text{GRCN}_{s/l}$ on *Office* in terms of Recall. The reason might be that it also suffers from the data sparsity problem when jointly training the extractor with recommendation models. The SGFD_{GRCN} outperforms all variants over all datasets, which demonstrates the effectiveness of our proposed SGFD.

2.5.3 Performance Comparison (RQ2)

We report the performance of three multimodal recommendation models equipped with our proposed SGFD and all the competitors over the three datasets in table 2.2. Our proposed approach achieves significant performance improvements over all the competitors across all the datasets in terms of different metrics. From the experiment results, we achieve some compelling observations.

Among all the competitors, NeuMF, CML, NGCF, and DGCF only use user-item interactions for recommendation. NeuMF could achieve promising performance compared with the classical linear interactions method [57]; this is attributed to the use of deep neural networks to model the non-linear interactions between users and items. Benefiting from the high-order information, NGCF and DGCF accomplish the state-of-the-art result. Both NGCF and DGCF exploit high-order connectivity between users and items through embedding propagation over the graph structure. DGCF outperforms NGCF by leveraging disentangled representation techniques to capture users' diverse intents, resulting in robust and independent user and item embeddings. However, its reliance on the dot product, which does not adhere to the triangle inequality, limits its ability to effectively model fine-grained user preferences [63]. In contrast, CML utilizes a metric-based learning approach [6], enabling it to surpass both NeuMF and NGCF by a significant margin.

[6] Metric-based learning approaches, such as those using distance functions (e.g., Euclidean distance), satisfy the triangle inequality. This property enables the model to more effectively capture relationships between embeddings while maintaining consistency within the representation space.

Table 2.2: Performance of our SGFD model and the competitors over three datasets. We categorize the methods into two categories (i.e., blocks): 1) methods in the 1st block are the ones relying only on the user-item interaction information without any side information, and 2) methods in the 2nd and 3rd block are the ones that consider both text and image features. Notice that the values are reported by percentage with '%' omitted.

Datasets	Office		Clothing		Toys&Games		Sports	
Metrics	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG
NeuMF	6.05	3.37	1.89	0.80	2.53	1.28	3.30	1.57
CML	12.29	13.95	4.09	2.24	12.27	11.60	9.82	6.40
NGCF	8.14	4.79	4.66	2.16	9.70	5.87	7.07	3.37
DGCF	12.06	11.05	7.45	4.05	12.62	10.85	10.26	6.29
GRCN	13.20	15.33	7.63	4.12	11.85	11.23	10.19	6.65
SGFD$_{GRCN}$	**13.88***	**16.26***	**8.16***	**4.35***	**12.59***	**11.93***	**10.91***	**6.89***
Improv.	5.15%	6.07%	6.95%	5.58%	6.24%	6.23%	7.07%	3.61%
BM3	13.67	15.36	7.43	4.21	12.22	11.92	10.95	5.20
SGFD$_{BM3}$	**14.07***	**15.86***	**7.84***	**4.57***	**12.54***	**12.94***	**11.17***	**5.51***
Improv.	2.93%	3.25%	5.52%	8.56%	2.62%	8.56%	2.01%	5.96%

The symbol * denotes that the improvement is significant with $p - value < 0.05$ based on a two-tailed paired t-test.

GRCN, and BM3 explore multimodal features, including textual and visual features, to learn better user and item representations for improving recommendation performance. Overall, the methods that consider multimodal features yield better performance than those that rely only on user-item interactions in most cases. Specifically, by discovering and pruning potential false-positive edges, GRCN outperforms all above-mentioned recommendation models in general. However, it underperforms DGCF on $Toys\&Games$, due to the limitation of entangled representations. The competitive performance of DGCF further demonstrates the benefits of using disentangled learning in recommendation. BM3 consistently achieves better performance than all the other baselines over all the datasets. It should be credited to the joint optimization of three multi-modal objectives to learn the representations of users and items.

To demonstrate the effectiveness of our proposed method, we equip GRCN and BM3 with SGFD, which are denoted as SGFD$_{GRCN}$ and SGFD$_{BM3}$, by employing the feature extractor in student model to replace the original feature extractor in the recommendation models. Note that our approach focuses on distilling knowledge from the teacher to the student model. Hence, the users and items representation learning approaches of the base models (GRCN and BM3) are not modified. The performance of SGFD$_{GRCN}$ and SGFD$_{BM3}$ are reported in Table 2.2. From the results, we can see that both SGFD$_{GRCN}$ and SGFD$_{BM3}$ obtain a significant im-

Table 2.3: Performance of our proposed SGFD method equipping on GRCN and their variants over three datasets. The best results are highlighted in bold. Notice that the values are reported by percentage with '%' omitted.

Datasets	Office		Clothing		ToysGames		Sports	
Metrics	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG
SGFD$_l$	13.30	15.80	7.69	4.06	12.01	11.31	10.64	6.72
SGFD$_f$	13.53	15.71	7.86	4.18	11.55	10.92	10.74	6.87
SGFD$_e$	13.71	15.73	7.97	4.28	12.08	11.36	10.80	6.75
SGFD$_s$	13.76	16.07	8.02	4.30	12.22	11.50	10.83	6.80
SGFD	**13.88**	**16.26**	**8.16**	**4.35**	**12.59**	**11.93**	**10.91**	**6.89**

provement over all datasets. We credit this to the combined effects of the following three aspects. Firstly, the teacher model can extract semantic-aware features from generic multimodal features. Secondly, the knowledge encoded in the teacher model is transferred to the student model by the response- and feature-based knowledge distillation losses. Thirdly, the feature extractors with shallow neural networks in the student model replace the original feature extractors in the base models. Based on this, existing multimodal models can ease the negative impact of the data sparsity problem on representation learning.

2.5.4 Ablation Study (RQ3)

In this section, we examine the contribution of different components to the performance of our method, which is equipped with the recommendation method (GRCN). Our analyses are based on performance comparisons with the following variants of our method.

- **SGFD$_l$**: It is a variant of SGFD that uses only logits-based distillation loss for knowledge distillation.
- **SGFD$_f$**: It is a variant of SGFD that uses only feature-based distillation loss for knowledge distillation.
- **SGFD$_e$**: This variant removes the complementary information exploited from multimodal features in the teacher model.
- **SGFD$_s$**: This variant adopts the shallow neural network as the feature extractor in the teacher model.

The variants SGFD$_l$ and SGFD$_f$ employ the logits-based distillation method and feature-based distillation method to transfer the knowledge encoded in the teacher extractor to the student extractor, respectively. Notice that, the SGFD$_l$ and SGFD$_f$ gain different improvements over three datasets, which indicates the benefits from the response-based knowledge and the feature-based knowledge are different. For

example, as the label information is messy and the visual feature contains more information about the characteristics of items, SGFD$_f$ yields better performance than SGFD$_l$ over $Clothing$. In other words, semantic information is more important on $ToysGames$.

SGFD$_e$ adopted both the logits- and the feature-based distillation loss to transfer the knowledge from the teacher extractor to the student extractor. Compared with SGFD$_l$ and SGFD$_f$, SGFD$_e$ achieves a remarkable performance across all the datasets in terms of Recall. However, the SGFD$_e$ underperforms SGFD$_l$ and SGFD$_f$ over $ToysGames$ and $Sports$ in terms of NDCG. The reason might be that the semantic information from labels and features is inconsistent. SGFD$_s$ outperforms SGFD$_e$ across all datasets in terms of NDCG. This is because the complementarity between different modality features is taken into account during the feature extraction process.

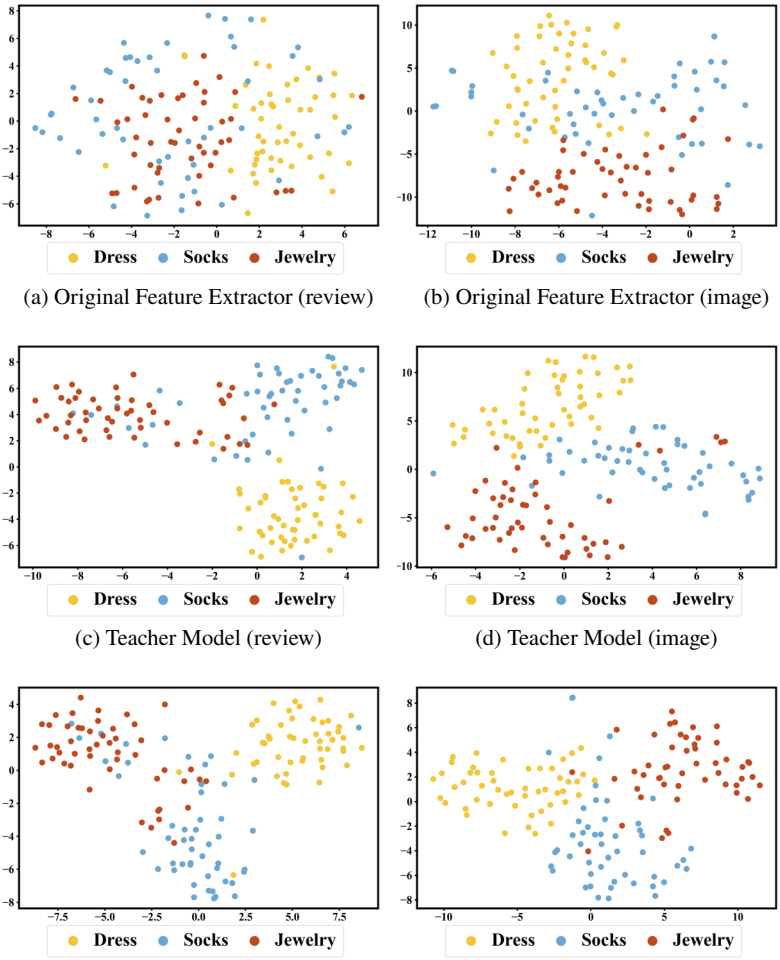
Our proposed SGFD method achieves superior performance across all datasets. It demonstrates the effectiveness of using the feature extractor with deep neural networks to extract the semantic-aware feature. Besides, the use of complementary information can also alleviate the inconsistent problem referred to above.

2.5.5 Visualization of Knowledge Transfer (RQ4)

To confirm the transfer of knowledge from the teacher model to the student model, we visualized the review feature and image feature vectors extracted by the original feature extractor in GRCN as well as the feature extractors in the teacher and student model of SGFD$_{GRCN}$, respectively.

In Fig. 2.5, the first three figures on the left depict the feature vectors derived from the generic review features of items, while the three figures on the right illustrate the vectors obtained from the generic image features of items. These items are randomly sampled from three categories of *Clothing* (accessories, body jewelry and socks). Each dot in the figures denotes the extracted vector from the feature of an item. Besides, the dots with the same color in all figures denote the same category of items. The obtained vectors are clustered and visualized via the t-Distributed Stochastic Neighbor Embedding (t-SNE).

As shown in Fig. 2.5, for the feature vectors extracted from different extractors, those items of the same category from the teacher extractor are closer than the original extractor, and the boundaries between categories are also more straightforward in Fig. 2.5(c). This is attributed to using semantic information to guide the feature extraction process in the teacher model. Besides, the clustering results shown in Fig. 2.5(e) demonstrate that the knowledge encoded in the teacher model is transferred to the student model. A similar trend can be observed from the clustering result of feature vectors extracted from generic image features based on the three extractors.



(a) Original Feature Extractor (review)

(b) Original Feature Extractor (image)

(c) Teacher Model (review)

(d) Teacher Model (image)

(e) Student Model (review)

(f) Student Model (image)

Fig. 2.5: Visualization of the vectors extracted from generic review and textual features. The vectors are derived from the original feature extractor in GRCN, feature extractors in teacher and student models of SGFD$_{GRCN}$. In addition, the items are sampled from three categories of *Clothing*: Socks (Orange), Dress (Blue) and Jewelry (Green).

2.6 Conclusion

In this chapter, we propose a novel model-agnostic approach, named Semantic-Guided Feature Distillation (SGFD), which can robustly extract effective and robust recommendation-oriented features from generic modality features for recommenda-

tion. In our approach, feature extractors in the teacher model are proposed to extract effective features considering both the semantic information of items from labels and the complementary information of multiple modalities. Moreover, to reduce the negative effect of the data sparsity problem on feature extraction in recommendation, we employ the student model consisting of feature extractors with shallow neural networks. By transferring the knowledge encoded in the teacher model to the student model and replacing the original feature extractor in recommendation methods with the feature extractors in the student model of our SGFD, effective recommendation-oriented features can be obtained. The experimental results on three real-world datasets demonstrate our approach can significantly improve the recommendation accuracy of multimodal recommendation methods.

Chapter 3
User Diverse Preference Modeling by Multimodal Attentive Metric Learning

3.1 Introduction

In Chapter 2, we explored the method for extracting effective multimodal features suitable for recommendation, which encompass rich item characteristics and reflect user preferences. Building on this, this chapter aims to demonstrate how to leverage these multimodal features to effectively capture users' diverse preferences, thereby addressing the limitations of existing methods.

Existing approaches primarily utilize user-item interactions to capture user preferences. For instance, in the widely used matrix factorization (MF) method [83, 9], given the user-item interaction matrix, all users and items are first mapped into a latent feature space, in which each user and item is presented as a feature vector. The preference of a user (u) for an item (i) is predicted based on the dot product of the feature vectors, i.e., $\boldsymbol{p}_u^T \boldsymbol{q}_i$. Many recommendation methods have been proposed based on this idea, like WRMF [66], BPR [134], and PMF [136]. In recent years, the matrix factorization for recommendation has been greatly advanced with the development of deep neural network (DNN) techniques. A set of DNN based matrix factorization methods has been proposed, such as the neural collaborative filtering (NCF) [57] and Deep MF [193].

Although great progress has been achieved thus far, most existing recommender systems have not well considered to characterize user varying preferences for different items. It is common that the user preference on the various aspects of items is different. For example, a user will value the "plot" more for a suspense movie, while pays more attention to the "special effects" for a super-hero movie. As shown in Fig. 3.1, we can see that users do not treat aspects equally for different items, even when the items are of the same category. Most existing recommendation methods use *the same vector to represent a user's preferences for all items*, which cannot accurately predict the diverse preferences on various items. Recently, some researchers have noticed this problem and proposed several models to tackle it. One type of methods leverages reviews to analyze user attention on different aspects of items and then integrates the results into the matrix factorization methods for recommen-

F. Liu et al., *Multimodal Learning toward Recommendation*,
https://doi.org/10.1007/978-3-031-83188-1_3

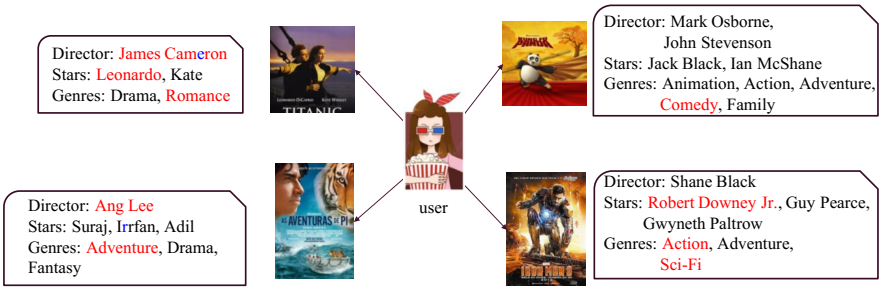


Fig. 3.1: An Example of User Diverse Preferences. Users often focus on different aspects when considering various items. The highlighted directors, stars, or genres represent the movie elements that attract more attention from the users.

dation, such as ALFM [27], A^{3}NCF [26], and ANR [30]. Another type of methods varies the target item or user vector based on the vectors of those most influential items or users with respect to the target items. These recently proposed methods are based on deep neural networks, such as deep interest networks (DIN) [217], memory attention-aware recommender system (MARS) [1], and collaborative memory network (CMN) [39].

With the consideration of users' varying preferences towards different items, the above methods have achieved better results over their baselines. However, these methods still rely on the MF method, and the dot product used in MF for similarity prediction does not satisfy the triangle inequality[1] [63, 200], meaning it is not a metric-based distance learning. This problem significantly limits the expressiveness of MF and hence causes MF-based methods incapable of capturing the fine-grained user preferences, resulting in sub-optimal performance [63, 209]. To tackle this problem, Hsieh et al. [63] proposed a metric collaborative filtering (CML) method which learns the preferences by minimizing the distance between user and item vectors (i.e., $||\boldsymbol{p}_u - \boldsymbol{q}_i||$) of positive interactions. However, directly applying of this distance metric is problematic [165]. Intuitively, this method tries to map each user and item pair with positive interaction to the same point in a low dimensional space. The problem is that each user and item has many interacted items and users, respectively. As a result, it is geometric inflexible for a large dataset since it tries to fit a user and all the interacted items into the same point. Tay et al. [157] has mathematically proved this is *geometrically restrictive* and will lead to an *ill-posed algebraic* system. They proposed to learn an adaptive relation vector r_{ui} between the interactions of a user-item pair by minimizing $||\boldsymbol{p}_u + \boldsymbol{r}_{ui} - \boldsymbol{q}_i||$. Although those proposed metric-based learning approaches tackled the problem of dot product, they have not well modeled user diverse preferences for various items.

Most existing recommender systems represent a user's preference with a feature vector, which is assumed to be fixed when predicting this user's preferences for

[1] It is defined as: "the distance between two points cannot be larger than the sum of their distances from a third point" [160].

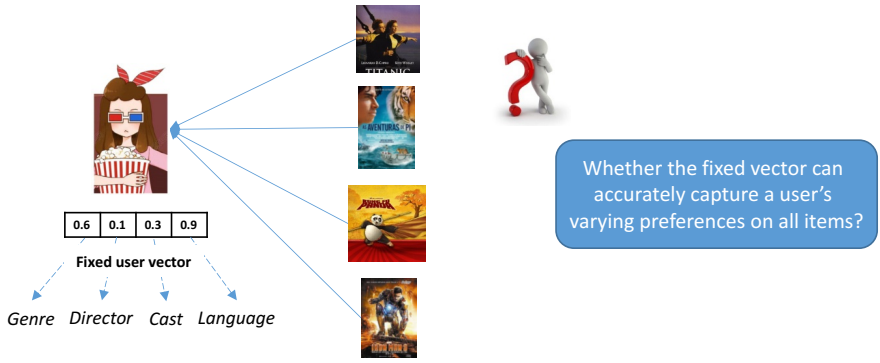


Fig. 3.2: Limitations of Traditional Methods in Representing a User's Preference with a Fixed Vector.

different items. However, the same vector cannot accurately capture a user's varying preferences on all items, especially when considering the diverse characteristics of various items. As shown in Fig. 3.2, the fixed user vector is used to predict the user's preferences for four movies of different genres. Based upon the aforementioned insights, in this chapter, we propose leveraging multimodal features to model user diverse preferences. Multimodal features contains rich item characteristics, and effectively utilizing this can help us use *different vectors to represent a user's diverse preferences for different items*. Specifically, we present a novel multimodal attentive metric learning (MAML) method for capturing users' diverse preferences. Our MAML model enhances the CML method by introducing a weight vector to each user-item pair. This weight vector represents the user's attention on different aspects of the target items, and thus it is unique for each user-item pair. This is achieved by a designed attention neural network which analyzes user attention on the target item by exploiting the textual and visual features of this item. The learned attention vector is then integrated with the user and item vectors to compute the user and item distance. In this way, our model can capture a user's varying preferences for different items in recommendation. It is also worth mentioning that our proposed MAML enjoys the following merits: 1) It satisfies the inequality property because of the adopted metric-based learning approach, and thus avoids the problem of the dot product similarity prediction method; And 2) it tackles the geometrically restrictive problem in CML. This is because the weight vector works as a transformation vector which projects the user and item into a distinct space (because the weight vector is unique for each user-item pair) to perform the proximate calculation via the Euclidean distance. Therefore, MAML allows a greater extent of geometric flexibility and modelling capability. Besides, we find that the standard attention mechanism is inferior to MAML because it leads to very small Euclidean distance between each dimension of the user and item vectors, which significantly diminish the prediction power of our model (see Sec. 3.4.2 for detail). To address this, we adjust the attention design by enlarging the summation value of the attention weights. We conduct com-

prehensive experiments on three benchmark datasets to evaluate the performance of our model on the top-*n* recommendation task. Experimental results demonstrate that our MAML outperforms the state-of-the-art MF-based and metric-based learning approaches. Beyond the improvement in accuracy, an exciting property of MAML is that it can uncover users' varying preferences on items.

The main contributions of MAML can be summarized as:

- We presented a novel Multimodal Attentive Metric Learning model for top-n recommendation. The proposed model can model users' diverse preferences on different aspects of items.
- We designed an attention network, which exploits the multimodal features of the target item to estimate a user's specific attention on each aspect of this item.
- Our model is developed based on a metric-based learning approach, which avoids the inherent limitation of matrix factorization-based methods and thus can capture fine-grained user preference.
- We validated the effectiveness of our model through extensive experiments conducted on four real-world datasets.

The rest of this chapter is organized as follows: Section 3.2 provides a comprehensive review of related work. Section 3.3 introduces the notions and background relevant to our proposed model. Section 3.4 details the components of the proposed model. Experimental results are presented in Section 3.5. Finally, Section 3.6 concludes the chapter.

3.2 Related Work

As a comprehensive review on recommendation techniques is far beyond the scope of this chapter, we mainly focus on the most related studies falling into the following three categories.

3.2.1 Diverse Preference Modeling

In recent two years, researchers have paid more attention to model the varying preferences towards different items and proposed several methods. We roughly categorize those methods into two groups. 1) Methods in the first group exploit reviews to analyze each user's attention on different aspects of the target item, and then integrate the attention weights into the matrix factorization for recommendation [27, 26, 30, 21]. In particular, ALFM [27] applies a topic model on reviews to detect the user attention, and A^3NCF [26] uses a neural attention network to learn the user attention from reviews. Following the same idea, Chin et al. [30] developed an end-to-end attentive neural network based recommendation model, which leverages reviews and ratings to learn user diverse preferences on different aspects of items. The AFM method [21]

adopts a similar strategy and exploits other types of side information (e.g., item category) instead of reviews. And 2) as to the methods in the second group, they adapt the user or item vector according to the most influential users or items, which are selected based on the target item. For example, in CMN [39], the target user vector is computed based on the neighbour user vectors, where the neighbour users and their contributions to the target user vector depend on the target item. MARS [1] adapts the target user vector based on the neighbour item vectors of the target item. Both CMN and MARS fix the vector of the target item while adapt the user vector. In contrast, DIN [217] fixes the user preference vector while adapts the target item representation based on the user's previously purchased items.

All the aforementioned methods are based on the matrix factorization framework by using dot product to estimate the similarity between the target user and item. With a different strategy in this work, we developed a novel user diverse modeling method based on the metric learning method, which automatically avoids the limitation of dot product. To the best of our knowledge, this is the first work to adopt metric learning method to capture user diverse preferences on different items.

3.2.2 Metric-based Learning

Several efforts have been dedicated to incorporating metric-based learning methods into collaborative filtering. Typically, the Euclidean embedding is the most popular one. For instance, Khoshneshin et al. [75] used the Euclidean embedding to model explicit feedback. Bachrach et al. [5] developed a method to transform the pre-trained dot-product space into a Euclidean space by preserving the item similarity. The most related studies are the recently proposed collaborative metric filtering (CML) [63] and latent relational metric learning (LRML) methods [157]. CML learns the user vector and item vector by minimizing the Euclidean distance of each user-item pair. Despite its simplicity, it can effectively encode user preference, as well as user-user and item-item similarity, achieving very competitive performance on several datasets. In [157], authors pointed out that CML is geometrically restrictive and leads to an ill-posed algebraic system. This is because it tries to map each pair of user-item into the same point in the Euclidean space, and there are many neighbours for each user and item. To tackle this problem, they introduced a relational vector to connect the user and item in the embedding space.

In this chapter, we introduce an adaptive weight vector to transform each user-item pair into a unique space (i.e., the user-item specific attentive-aspect space) for minimization. Therefore, our method could also deal with the problem of CML. Besides, the weight vector is tailored for each pair of user-item, and thus our method can capture users' varying preferences upon items.

3.2.3 Multimodal User Preference Modeling

Although many recommender systems have been designed to exploit side information to enhance the recommendation performance, most of them only exploit a single modality feature, such as textual reviews [15, 121, 155, 26, 16, 27, 30], item images [73, 168, 54, 122], and other metadata [21, 24, 45, 130]. More recently, several deep models have been proposed to model user preference by using multimodal information. Zhang et al. [204] proposed a knowledge base method, which extracts the multimodal knowledge as well as unstructured textual and visual knowledge to jointly learn the latent representations of items within a collaborative filtering framework. Similarly, Zhang et al. [211] first extracted user and item features from ratings, reviews, and images via deep representation learning and then concatenated the multimodal features to form the joint representations of users and items for recommendation.

Our proposed method is different from the above methods from two perspectives. Firstly, the above methods leverage the multimodal features to jointly learn fixed user and item representations for recommendation, while our method exploits the multimodal features to learn user's varying attention to the aspects of different items. Secondly, our method is based on a metric learning strategy and adopts the Euclidean distance to predict the user preference on an item, which is greatly different from the methods relying on the matrix factorization framework and the dot product for similarity estimation.

3.3 Notations and Background

3.3.1 Notations

Before describing of our model, we would like to first define some basic notations. Let $\mathcal{U}$ be the user set, $\mathcal{I}$ be the item set, and $\boldsymbol{R}$ be the user-item interaction matrix. $r_{u,i} \in \boldsymbol{R}$ indicates the interaction between the user $u \in \mathcal{U}$ and the item $i \in \mathcal{I}$. The interaction could be implicit or explicit. For implicit feedback, $r_{u,i} = 1$ if the interaction between u and i exists; otherwise, $r_{u,i} = 0$. For explicit feedback, $r_{u,i}$ usually is the rating that u assigns to i. Let $\mathcal{R}$ be the set of user-item (u, i) pairs whose values are non-zero and $\mathcal{R}_u$ be the set of items that have been interacted by u. $\boldsymbol{p}_u \in \mathbb{R}^f$ and $\boldsymbol{q}_i \in \mathbb{R}^f$ denote the latent feature vector for u and i, respectively.[2] The core of recommender systems is to learn the latent feature vectors of users and items (i.e., $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$). When recommending items to a user u, for each item $j \notin \mathcal{R}_u$, its recommendation score $\hat{r}_{u,j}$ is computed based on $\boldsymbol{p}_u$ and $\boldsymbol{q}_j$.

[2] In this chapter, unless otherwise specified, notations in bold style denote matrices or vectors, and the ones in normal style denote scalars.

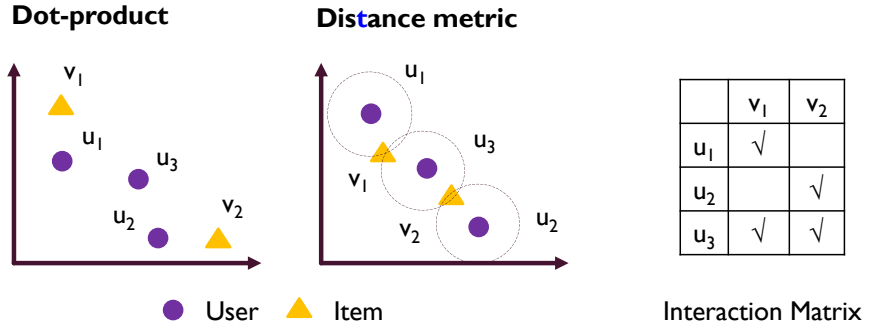


Fig. 3.3: Comparison of Dot Product and Distance Metric.

3.3.2 Background

3.3.3 Overview

Matrix factorization (MF) [83] maps users and items into a shared latent feature space and estimates an unseen interaction by the dot product between the user and item vector, i.e., $\hat{r}_{u,i} = \boldsymbol{p}_u^T \boldsymbol{q}_i$. And $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$ for all $u \in \mathcal{U}$ and $i \in \mathcal{I}$ are learnt by minimizing the predicted errors between the $\hat{r}_{u,i}$ and $r_{u,i}$, namely, $\sum_{u,i \in \mathcal{R}} \|\hat{r}_{u,i} - r_{u,i})\|^3$. Due to its simplicity and superior performance, matrix factorization has become the most popular collaborative filtering method over the past decade. The original MF models were designed for rating prediction [83, 57] and was extended to weighted regularized matrix factorization (WRMF) for implicit feedback prediction [66]. As the goal of recommender system is to provide the target user with a ranking list of top few items, which are most likely to be preferred by the user, recommendation is rather a ranking problem instead of rating. In light of this, MF moves to model the relative preferences between different items and the pairwise learning approach has been widely adopted to achieve the goal [134, 211]. In pairwise learning, the user and item vectors are learnt by setting $\hat{r}_{u,i} > \hat{r}_{u,k}$ for any two pairs that satisfy $(u,i) \in \mathcal{R}$ and $(u,k) \notin \mathcal{R}$. A typical example is the bayesian personalized ranking (BPR) [134]. It is a MF based method, in which $\hat{r}_{u,i} = \boldsymbol{p}_u^T \boldsymbol{q}_i$ and $\hat{r}_{u,k} = \boldsymbol{p}_u^T \boldsymbol{q}_k$ and thus $\boldsymbol{p}_u^T \boldsymbol{q}_i > \boldsymbol{p}_u^T \boldsymbol{q}_k$ (s.t. $(u,i) \in \mathcal{R}$ and $(u,k) \notin \mathcal{R}$) is used for optimization.

Despite the success of MF, as illustrated in Fig. 3.3, it is not a metric-based learning approach as the adopted dot product similarity does not satisfy the triangle inequality property, which is critical for modeling fine-grained user preference as demonstrated in [63, 209]. From the metric learning perspective, since users and items are represented as latent vectors in a shared space, the similarity between a user and an item can be estimated based on the Euclidean distance between their vectors as:

[3] Notice that here we ignore the regularization term for the simplicity of presentation.

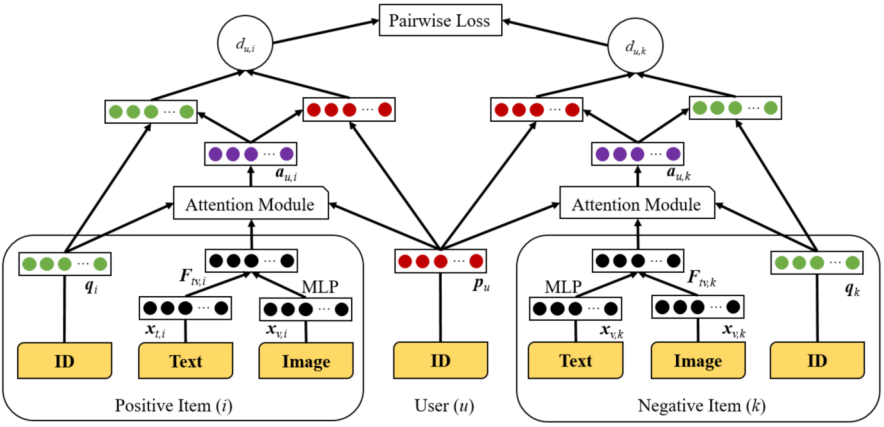


Fig. 3.4: Overview of our MAML model. For each user-item pair, we propose an attention neural network, which exploits the item's multimodal features to estimate the user's special attention to different aspects of this item. The obtained attention is then integrated into a metric-based learning method to predict the user preference on this item.

$$d(u,i) = \|\boldsymbol{p}_u - \boldsymbol{q}_i\|. \tag{3.1}$$

Considering the pairwise learning, it is natural to derive that $d(u,i) < d(u,k)$ for $(u,i) \in \mathcal{R}$ and $(u,k) \notin \mathcal{R}$, where i denotes an item that u likes and k denotes an item that u does not like. The collaborative metric learning (CML) [63] is designed based on this simple idea. As this method is a metric-based learning approach, it naturally avoids the limitation of dot product and achieves better performance than the MF models [63]. Therefore, in this work, we also derive our user diverse preference modeling method based on this approach instead of MF.

3.4 Multimodal Attentive Metric Learning

3.4.1 Overview

In the aforementioned methods, they all utilize a fixed vector $\boldsymbol{p}_u$ to represent a user u's preference in the feature space. In those models which map users and items into a joint latent space for similarity estimation, they all assume that each dimension in the space stands for a type of feature or an aspect of the items. Those dimensions/aspects are expected to well describe and distinguish the preferences of different users. Based on this understanding, we argue that using the same vector $\boldsymbol{p}_u$ to predict u's preferences for all items may be not optimal, because in the real scenarios, it is common that the preference of a user on the aspects of different items is varying. For

example, a user who prefers the *taste* and *price* of a restaurant may pay more attention to the *ambience* and *service* of another restaurant, because the two restaurants serve for different purposes. When predicting the preference of a user u towards an item i, those aspects of the item i to which u pays the most attention should dominate u's preference on the item i.

In light of this, we propose a multimodal attentive metric learning (MAML) model. For each user-item (u, i) pair, our model computes a weight vector $\boldsymbol{a}_{u,i} \in \mathbb{R}^f$ to indicate the importance of i's aspects for u. In addition, the side information of items is exploited to estimate the weight vector, as side information conveys rich features of items, especially text reviews and item images, which are well-recognized to provide notable and complementary features of items in different aspects [211, 25]. We adopt the recent advancement of attention mechanism [20, 26] to estimate the attention vector. With the attention (weight) vector, the Euclidean distance between a user u and an item i in our model becomes:

$$d(u, i) = \|\boldsymbol{a}_{u,i} \odot \boldsymbol{p}_u - \boldsymbol{a}_{u,i} \odot \boldsymbol{q}_i\|, \tag{3.2}$$

where $\odot$ denotes the element-wise product between vectors.

It is worth mentioning that, with the attention vector, our model can not only accurately capture the user's varying preferences for different items, but also tackle the geometrically restrictive problem [157] in CML. From Eq. 3.1, it can be found that CML tries to fit a user and all the interacted items into the same point in the latent space, however, each item in turn has many interacted users. Therefore, it is geometrically impossible to achieve the goal. In our model, as $\boldsymbol{a}_{u,i}$ is unique for each user-item pair, it works as a transformation vector which transforms the target user and item into a new space for distance computation. Thus, our method can naturally avoid the geometrically restrictive problem in CML. An overview of our proposed modal MAML is illustrated in Fig. 3.4.

We adopt the pairwise learning for optimization and the loss function is defined as:

$$L_m(d) = \sum_{(u,i)\in R} \sum_{(u,k)\notin R} \omega_{ui}[m + d(u, i)^2 - d(u, k)^2]_+, \tag{3.3}$$

where i is an item that u likes and k is an item that u does not like; $[z]_+ = max(z, 0)$ denotes the standard hinge loss. ω_{ui} is a ranking loss weight (described in Sect. 3.4.3) and $m > 0$ is the safety margin size. $d(u, i)$ and $d(u, k)$ are computed according to Eq. 3.2. In the next, we introduce how to compute the attention vector $\boldsymbol{a}_{u,i}$ for each user-item pair.

3.4.2 Attention Mechanism

In this section, we introduce the attention mechanism in MAML for capturing a user u's specific attention $\boldsymbol{a}_{u,i}$ of an item i. Since text reviews and images contain rich information about user preference and item characteristic, they are used to capture

u's attention on the various aspects of i. A two-layer neural network is used to compute the attention vector:

$$\boldsymbol{e}_{u,i} = Tanh(\boldsymbol{W}_1[\boldsymbol{p}_u; \boldsymbol{q}_i; \boldsymbol{F}_{tv,i}] + \boldsymbol{b}_1), \tag{3.4}$$

$$\hat{\boldsymbol{a}}_{\boldsymbol{u},\boldsymbol{i}} = \mathbf{v}^T ReLU(\boldsymbol{W}_2\boldsymbol{e}_{u,i} + \boldsymbol{b}_2), \tag{3.5}$$

where $\boldsymbol{W}_1, \boldsymbol{W}_2$ and $\boldsymbol{b}_1, \boldsymbol{b}_2$ are respectively the weight matrices and bias vectors of the two layers. $\mathbf{v}$ is a vector that projects the hidden layer into an output attention weight vector. $\boldsymbol{F}_{tv,i}$ is the item feature vector which is a fusion of i's textual feature and image feature (described later). $[\boldsymbol{p}_u; \boldsymbol{q}_i; \boldsymbol{F}_{tv,i}]$ denotes the concatenation of $\boldsymbol{p}_u$, $\boldsymbol{q}_i$, and $\boldsymbol{F}_{tv,i}$. Tanh and ReLU [147, 109, 110] are used as the activation functions for the first and second layer, respectively.[4]

Following the standard procedures of neural attention networks, there is a subsequent step to normalize $\hat{\boldsymbol{a}}_{\boldsymbol{u},\boldsymbol{i}}$ with the softmax function, which converts the attention weights to a probabilistic distribution. Unfortunately, this standard solution does not work well in practice. This is because in our model, the attention weights are directly used to element-wise product with the Euclidean distance between $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$ (see Eq. 3.2). For each dimension l, the weighted distance is $d_{u,i,l} = a_{u,i,l} \cdot (p_{u,l} - q_{i,l})$. After softmax normalization, the weights will be very small. For example, when the dimension f is 100, the mean value of weights is only 0.01. Notice that the distance between each dimension of $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$ is already quite small [5]. With such a small weight $a_{u,i,l}$, the distance $d_{u,i,l}$ becomes even smaller. When the distances of all dimensions are quite small, the differences between different dimensions (aspects) become negligible. This will weaken the distinguishing power of our model, resulting in performance deterioration. To alleviate this problem, we propose to enlarge the normalized weight by a factor α. In our model, the final attention weight vector is computed as:

$$a_{u,i,l} = \alpha \cdot \frac{exp\left(\hat{a}_{u,i,l}\right)}{\sum_{l=1}^{f} \hat{a}_{u,i,l}}. \tag{3.6}$$

In experiments, we set α to be the dimension of the weight vector, i.e., $\alpha = f$, which turns out to work well (see Section **??**). This setting is motivated by the consideration that when the weight vector is binary, only the aspects with the weight 1 take effect on the final decision, and the extreme case is that all the aspects are of the same importance. In the next, we introduce how to obtain the fused item feature vector $\boldsymbol{F}_{tv,i}$.

Item Features

For each item, its textual and visual features are extracted from its associated reviews and images.[6] The text feature $\boldsymbol{F}_{t,i}$ is extracted by the PV-DM [85] model, which learns continuous distributed vector representations for textual documents in an un-

[4] We have empirically studied different combinations of the activation functions in the attention neural network, and find out that Tanh and Relu can achieve relatively better performance.

[5] Due to the regularization $\|\boldsymbol{p}_*\|^2 \leq 1$ and $\|\boldsymbol{q}_*\|^2 \leq 1$, see Sect. 3.4.4

[6] In this work, each item is associated with one image.

supervised way. It considers the word sequence information and tries to preserve the semantic features. PV-DM takes text documents as inputs and outputs their vector representations in a latent semantic space. The visual feature $\boldsymbol{F}_{v,i}$ is extracted by the Caffe reference model [71], which consists of 5 convolutional layers and 3 fully connected layers. This model was pre-trained on 1.2 million ImageNet (ILSVRC2010) images. In this work, we take the output of the second fully connected layer, resulting in a 4096-D feature vector as the visual features for each item.

After extracting the textual and visual features of items, we fuse them to better represent the characteristics of items [166]. Many multimodal feature fusion methods have been proposed and we adopt a widely used strategy[205, 148] by first concatenating the textual and visual features and then feeding them into a multiple-layer neural networks. Specifically, the feature fusion network is,

$$\begin{aligned} z_1 &= \sigma(\boldsymbol{W}_1[\boldsymbol{F}_{t,i}; \boldsymbol{F}_{v,i}] + \boldsymbol{b}_1), \\ z_2 &= \sigma(\boldsymbol{W}_2 z_1 + \boldsymbol{b}_2), \\ &\ldots\ldots, \\ z_L &= \sigma(\boldsymbol{W}_L z_{L-1} + \boldsymbol{b}_L), \end{aligned} \tag{3.7}$$

where $\boldsymbol{W}_l$ and $\boldsymbol{b}_l$ denote the weight matrix and bias vector for the l-th layer, respectively. $\sigma(\cdot)$ is the activation function and ReLU is adopted because its biologically plausible and non-saturated property [47]. The output of last layer is the fused feature, namely, $\boldsymbol{F}_{tv,i} = z_L$. Noticed that our focus in this chapter is to exploit items' multimodal features for capturing users' varying attentions on different aspects of various items. The above extraction and fusion method is adopted for simplicity, and other advanced feature extraction and fusion methods could also be adopted in this case.

3.4.3 Ranking loss weight

We adopt the Weighted Approximate-Rank Pairwise (WARP) loss [187] to compute $\omega_{u,i}$. This scheme penalizes a positive item at a lower rank much more heavily than the one at the top, and produces the state-of-the-art results in previous work [63, 214]. Given a metric d, let $rank_d(u,i)$ denote the position of i in u's recommended ranking list, $\omega_{u,i}$ is computed as,

$$\omega_{u,i} = log(rank_d(u,i) + 1). \tag{3.8}$$

In WARP, $rank_d(u,i)$ is estimated through a sequential sampling procedure which repeatedly samples negative items to find impostors [187]. For each user-item pair (u,i), let J denote the total number of items and M denote the number of impostors in N samples. The $rank_d(u,i)$ is approximated as $\lfloor (J \times M)/N \rfloor$. For more details on WARP, please refer to [187]. In our implementation, we follow the procedure described in [63] to estimate $rank_d(u,i)$.

3.4.4 Regularization

As text reviews and item images represent items' characteristics, we would like that items with similar textual and visual features to be closer in the latent feature vector. To achieve the goal, we define the following L_2 loss function,

$$L_f(q_*) = \sum_i \|\boldsymbol{F}_{tv,i} - \boldsymbol{q}_i\|. \tag{3.9}$$

This function penalizes an item i's feature vector $\boldsymbol{q}_i$ when $\boldsymbol{q}_i$ deviates away from the extracted feature $\boldsymbol{F}_{tv,i}$.

To prevent the redundant of each dimension in the feature space,we then employ another regularization technique, covariance regularization [31] to reduce the correlation between the activation in a deep neural network. This technique can be also used in our model to de-correlate the dimensions in the feature space and thus to maximize the utilization of the given space. Let $\mathbf{y}^n$ denote the latent vector of an object, which could be a user or an item; and n indexes the object in a batch of size N. The covariances between all pairs of dimensions i and j from a matrix C are defined as:

$$C_{i,j} = \frac{1}{N} \sum_n \left(y_i^n - \mu_i\right) \left(y_j^n - \mu_j\right), \tag{3.10}$$

where $\mu_i = \frac{1}{N} \sum_n y_i^n$. We define the loss L_c to regularize the covariances:

$$L_c = \frac{1}{N} \left(\|C\|_f - \|diag\,(C)\|_2^2\right), \tag{3.11}$$

where $\|\cdot\|_f$ is the Frobenius norm.

Finally, similar to the previous metric-based collaborative filtering methods [63, 157], we also bound all the user and item vector within a Euclidean unit sphere, i.e., $\|\boldsymbol{p}_*\|^2 \leq 1$ and $\|\boldsymbol{q}_*\|^2 \leq 1$, for regularization and preventing overfitting.

3.4.5 Optimization

With the consideration of all the regularization terms, the final object function of our MAML is,

$$\begin{aligned} &\min_{\theta, \mathbf{p}_*, \mathbf{q}_*} L_m + \lambda_f L_f + \lambda_c L_c \\ s.t. \quad &\|\mathbf{p}_*\|^2 \leq 1 \; and \; \|\mathbf{q}_*\|^2 \leq 1, \end{aligned} \tag{3.12}$$

where λ_f and λ_c are hyperparameters that control the weight of each loss term. The optimization is quite standard and the stochastic gradient descent (SGD) algorithm is adopted. In implementation, the Adam optimizer [77] is adopted to tune the learning rate.

Table 3.1: Basic statistics of the experimental datasets.

Dataset	#user	#item	#interactions	sparsity
Office	4,874	2,406	53,137	99.55%
Men Clothing	4,955	5,028	32,363	99.87%
Women Clothing	19,244	14,596	135,326	99.95%
Toys Games	18,748	11,673	161,653	99.93%

3.5 Experiments

To validate the effectiveness of our method, we conducted extensive experiments on four public datasets to answer the following research questions:

RQ1: Does our proposed MAML improve the performance of state-of-the-art recommendation models on the top-n recommendation task?

RQ2: How does the multimodal attention mechanism impact the performance of multimodal recommendation models?

RQ3: Can our method effectively capture users' diverse preferences to improve recommendation quality?

Next, we first introduce the experimental setup and then report and analyze the experimental results.

3.5.1 Experimental Setup

Datasets

The public Amazon review dataset [121], which has been widely used for recommendation evaluation in previous studies [211, 21, 25], is adopted for experiments in this chapter. We adopted four product categories as shown in Table 3.1. Since we need to exploit reviews and item images to extract the item features, items without any textual reviews or images were removed. After this step, we further pre-processed the dataset to keep only the items and users which have at least 5 interactions. The basic statistics of the four datasets are shown in Table 3.1. As we can see, the datasets are of different sizes and sparsity. For example, the *office* dataset is relatively denser than other datasets. The diversity of datasets is useful for analyzing the performance of our method and the competitors in different situations.

In this chapter, we focus on the top-n recommendation task, which aims to recommend a set of n top-ranked items that will be appealing to the target user. For each dataset, we randomly selected 70% of the interactions from each user to construct the training set, and the remaining 30% for testing. Each user has at least 5 interactions, thus we had at least 3 interactions per user for training, and at least 2 interactions per

user for testing.

Baselines

We compare our MAML method with the following baselines, including both shallow (BPR [134], VBPR [54]) and deep (DeepCoNN [215], NeuCF [57], JRL [211]) MF based models, as well as the metric learning based method CML [63]. Besides, they cover different information sources, i.e., ratings (BPR, CML), reviews (DeepCoNN, CML$_F$ [63], JRL), images (CML$_F$, JRL). Notice that CML$_F$ is an extension of CML. It extends the CML model by considering item features for top-n recommendation. We use CML$_{text}$, CML$_{image}$, and CML$_{all}$ to denote the model considering the text feature, image feature, and both features, respectively.

- **BPR** [134]: Bayesian Personalized Ranking [134] is a collaborative filtering method, which combines the matrix factorization method with a pair-wise learning to rank loss function. It has been proven to be a competitive baseline for this task [57, 25].
- **VBPR** [54]: This is a visual BPR method for top-n recommendation, which is the state-of-the-art method for recommendation based on visual images of the products.
- **NeuCF** [57]: The neural collaborative filtering method generalizes the matrix factorization to neural networks. It explores the non-linear interaction between user vectors and item vectors for top-n recommendation and achieves the state-of-the-art performance over the methods which only rely on user-item interaction information.
- **DeepCoNN** [215]: The Deep Cooperative Neural Networks model exploits the reviews of users and items in CNN to learn the user and item representations, which are then concatenated and passed into a regression layer for rating prediction.
- **JRL** [211]: The Joint Representation Learning model integrates reviews, product images and ratings to obtain the joint representations of users and items by using deep representations learning frameworks. It is a state-of-the-art deep model for the top-n recommendation.
- **CML** [63]: The Collaborative Metric Learning model learns a joint metric space to encode users' preferences and the user-user and item-item similarity for top-n recommendation.
- **CML$_F$** [63]: It extends the CML model by considering item features for top-n recommendation. We use CML$_{text}$, CML$_{image}$, and CML$_{all}$ to denote the model considering the text feature, image feature, and both features, respectively.

Evaluation Metrics

Four widely used evaluation metrics for top-n recommendation are used in our evaluation, including *precision*, *recall*, *NDCG* [69], and *Hit Ratio (HR)*. Below, we provide detailed explanations of *Precision* and *Hit Ratio*, as *Recall* and *NDCG* have already been introduced in Chapter 2.

Precision measures the proportion of recommended items that are relevant to the user. For a given user u, Precision is defined as:

$$\text{Precision} = \frac{|\text{Rel}(u) \cap \text{Rec}(u, n)|}{|\text{Rec}(u, n)|}, \tag{3.13}$$

where:

- $\text{Rel}(u)$ represents the set of relevant items for user u in the test set.
- $\text{Rec}(u, n)$ is the set of top-n recommended items for user u.

This metric evaluates the accuracy of the recommendation list by considering how many of the recommended items are actually relevant.

Hit Ratio measures whether at least one of the recommended items is relevant to the user. For a given user u, HR is defined as:

$$\text{HR} = \begin{cases} 1 & \text{if } \text{Rel}(u) \cap \text{Rec}(u, n) \neq \emptyset, \\ 0 & \text{otherwise.} \end{cases} \tag{3.14}$$

This metric is often aggregated across all users as the proportion of users for whom at least one relevant item appears in the top-n recommendations.

Note that all the metrics are computed based on the top 10 results. The reported results are the average values across all the testing users.

Experimental Settings

To ensure the consistency of baseline implementation, we adopted the publicly available implementations of BPR [7], VBPR [8], NeuCF [9], DeepCoNN [10], JRL [11], and CML [12]. We implemented our model with TensorFlow [13] and carefully tuned the key parameters. Specifically, we tuned the initial learning rate $\ell_0 \in \{0.1, 0.01, 0.001, 0.0001\}$; the margin m in the hinge loss $m \in \{0.1, 0.2, \ldots, 2.0\}$; the number of negative samples (s) and the regularization parameters $\{s, \lambda_f, \lambda_c\} \in \{1, 2, \ldots, 10\}$. In our experiments, the following setting works well: $\ell_0 = 0.001$, m =1.6 (1.5 for ToysGames), $s = 8$ (4 for Office and MenClothing). $\lambda_c = 5$, and $\lambda_f = 7$ (2 for ToysGames). The optimal batch size varies across different datasets. Besides, model parameters are saved in every 10 epochs and the models are trained in maximum 1,000 epochs. In experiments, the dimension of latent vectors (i.e., $\boldsymbol{p}_u$ and $\boldsymbol{q}_i$) of all methods is set to 64. Notice that all the competitors have also been carefully tuned for fair comparisons. For example, for CML, we found that $\lambda_c = 1$

[7] https://github.com/sh0416/bpr.

[8] https://github.com/arogers1/VBPR.

[9] https://github.com/hexiangnan/neural_collaborative_filtering.

[10] https://github.com/chenchongthu/DeepCoNN.

[11] https://github.com/evison/JRL.

[12] https://github.com/changun/CollMetric.

[13] https://www.tensorflow.org.

(all the other hyperparameters are the same as our MAML model) works better. We released the codes and involved parameter settings to facilitate others to repeat this work[14].

Table 3.2: Performance of our MAML model and the competitors over *Office* and *Men Clothing* datasets. Noticed that the values are reported by percentage with '%' omitted.

Datasets	Office				Men Clothing			
Metrics	NCDG	RECALL	HR	PR	NCDG	RECALL	HR	PR
BPR	4.557	7.780	21.410	2.562	1.229	2.375	4.611	0.478
NeuCF	4.145	6.609	19.504	2.516	1.416	2.599	6.017	0.625
CML	6.680	8.986	24.025	3.141	2.520	4.040	7.815	0.811
DeepCoNN	2.418	3.842	11.273	2.160	0.813	1.510	3.643	0.491
CML_{text}	6.889	9.171	24.778	3.223	3.212	4.952	9.659	1.005
MAML_{text}	7.094	9.346	25.165	3.222	3.282	4.961	9.843	1.019
VBPR	3.072	4.942	15.381	1.817	1.061	1.914	3.873	0.392
CML_{image}	6.987	9.232	24.560	3.214	3.542	5.259	10.419	1.086
MAML_{image}	7.077	9.286	25.155	3.261	3.585	5.401	10.927	1.148
JRL	3.391	3.421	12.345	1.525	2.574	3.920	7.247	0.657
CML_{all}	7.032	9.349	24.911	3.236	3.718	5.408	10.627	1.127
MAML_{all}	**7.139***	**9.386***	**25.177***	**3.265***	**3.733***	**5.574***	**10.719***	**1.152***

The symbol * denotes that the improvement is significant with $p - value < 0.05$ based on a two-tailed paired t-test.

3.5.2 Performance Comparison (RQ1)

The results of our model and all the competitors over the four datasets are reported in Table 3.2 and Table 3.3. It can be found that our method outperforms all the competitors consistently across all the test datasets in terms of different metrics. Besides, by grouping all the methods into four categories, we gained some interesting observations.

Firstly, we focused the performance of the methods in the first block, which only use the user-item interaction information. Specifically, NeuCF adopts neural

[14] https://github.com/liufancs/MAML.

Table 3.3: Performance of our MAML model and the competitors over *Women Clothing* and *Toys Games* datasets. Noticed that the values are reported by percentage with '%' omitted.

Datasets	Women Clothing				Toys Games			
Metrics	NCDG	RECALL	HR	PR	NCDG	RECALL	HR	PR
BPR	0.572	1.086	2.062	0.244	0.807	1.733	3.235	0.385
NeuCF	1.136	2.115	4.377	0.491	2.374	4.558	9.801	1.374
CML	3.026	4.542	9.157	1.034	6.207	8.638	17.454	2.204
DeepCoNN	0.648	1.214	2.474	0.394	1.496	2.581	5.665	1.145
CML_{text}	3.507	5.139	9.972	1.144	6.345	8.787	17.957	2.260
MAML_{text}	3.590	5.246	10.329	1.175	6.410	8.954	18.093	2.289
VBPR	0.921	1.644	3.357	0.377	0.846	1.595	3.738	0.418
CML_{image}	3.704	5.499	10.804	1.237	6.362	8.923	17.951	2.301
MAML_{image}	3.733	5.554	10.956	1.246	6.486	9.087	18.312	2.314
JRL	1.729	2.906	4.965	0.457	2.302	4.599	8.232	1.382
CML_{all}	3.798	5.562	10.985	1.241	6.409	8.928	18.135	2.308
MAML_{all}	**3.809***	**5.636***	**11.144***	**1.260***	**6.547***	**9.111***	**18.341***	**2.327***

The symbol * denotes that the improvement is significant with $p - value < 0.05$ based on a two-tailed paired t-test.

networks which can better model user-item interactions and achieve better results than BPR in general. The *Office* dataset has a smaller size but is relatively denser, which might be the reason that BPR can obtain good performance on this dataset.[15] CML outperforms both BPR and NeuMF by a large margin, because CML can capture fine-grained user preferences by using the metric-based learning approach. It can encode not only user preference but also the user-user and item-item similarity.

The second and third blocks are the methods exploiting one type of side information, i.e., text and image, respectively. MAML_{text} and MAML_{image} are variants of our model MAML, which exploits only text and image, respectively. As we can see, the recommendation performance can be greatly improved with the exploitation of additional item features, which has been demonstrated in many previous studies. DeepCoNN utilizes text information but performs unsatisfactorily. This is because the DeepCoNN uses text reviews not only in training but also in the test stage. In real scenario, the user review for an item is unavailable before the purchasing behavior happened. Therefore, in our experiments, we did not use review informa-

[15] Remind that MF based method suffers from the cold-start problem (i.e., very limited interactions for users or items) and NeuMF needs relatively more data for better learning.

tion in the testing stage, resulting in the performance degradation of DeepCoNN (which has also been demonstrated in [15]). It is interesting that the performance based on images outperforms that based on texts (by comparing CML_{image} and MAML_{image} to CML_{text} and MAML_{text}). This might because the image features are more important for the selected four types of products (e.g., *Clothing*). VBPR achieves satisfactory performance in *Women Clothing* but it is inferior to NeuCF and BPR on other datasets.

Finally, in the last block, the methods exploit both text and image features. The best performance of CML_{all} and MAML_{all} demonstrates the effectiveness of fusing multimodal features, which can indeed enhance the recommendation accuracy by considering multi-source item information. JRL achieves the best results over all the other baselines except the CML based ones (and BRP on the *Office* dataset). The CML based methods outperform all the other baselines, verifying the potential of metric-based learning approach. The only difference between MAML and CML is that the former models user diverse preferences by using the attention mechanism. The consistent and significant improvement of MAML over CML validates the importance of considering user diverse preferences on different aspects of items, and also demonstrates the effectiveness of our proposed model.

3.5.3 Effects of Our Attention Mechanism (RQ2)

Remind that in our MAML model, we adapt the standard attention mechanism by enlarging the sum of all the attention weights by a factor α. In our implementation, the value of α is set to be equal to the dimension of the user/item feature vectors, i.e., $\alpha = f$ (see Sect. 3.4.2 for details). In this section, we examine the effectiveness of this tailored attention mechanism in our model. In particular, we compare the performance of our model with the adapted attention mechanism (with α) and the standard one (without α). We tuned MAML with the standard attention mechanism carefully (e.g., $m \in \{0.001, 0.005, 0.01, \ldots, 2.0\}$, and $\{\lambda_f, \lambda_c\} \in \{0.0001, 0.001, \ldots, 10\}$). The best results are used for comparisons. As shown in Fig. 3.5, with the adapted attention mechanism, our model can achieve consistent improvement by a large margin across all the test datasets, indicating the effectiveness of our attention mechanism in MAML.

3.5.4 Visualization (RQ3)

In the proposed method, we model user diverse preferences by capturing user attention to different aspects of items. Our intuition is that the preference of a user on different aspects is varying when facing different items. Therefore, for each user-item pair, our model computes a unique attention weight for each aspect of the item (i.e., different dimensions of the feature vector). In this section, we would like to:

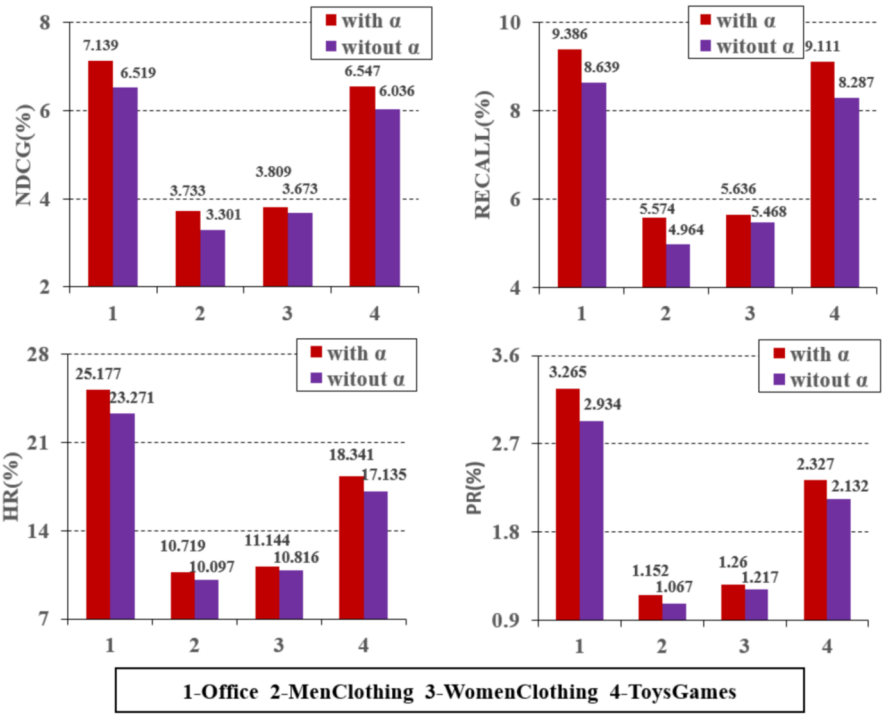
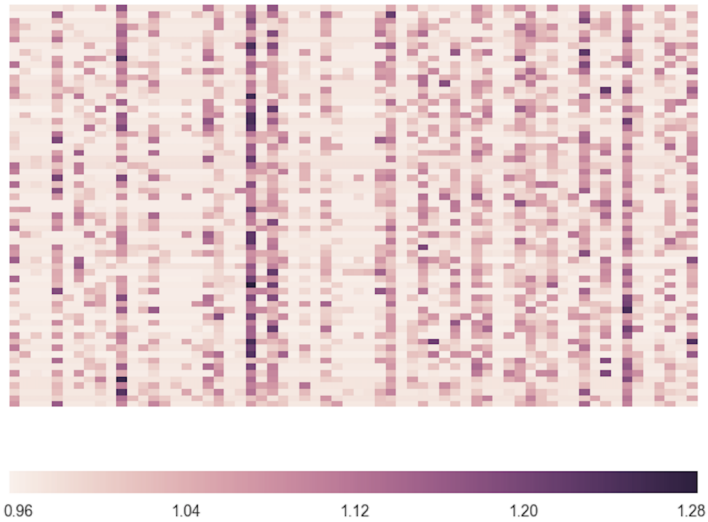


Fig. 3.5: Our adapted attention mechanism (with α) *vs.* the standard attention mechanism (whiteout α).

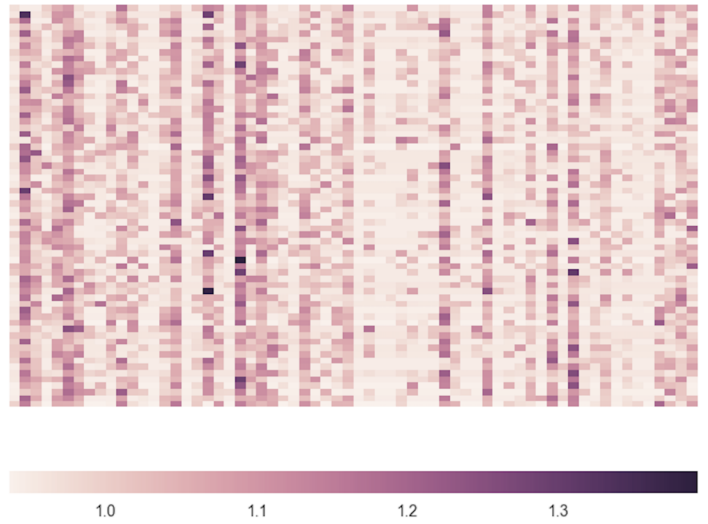
1) visualize the learned attention weights to illustrate user varying preferences on different aspects of various items; and 2) visualize user diverse preferences based on the purchased items.

Attention Visualization

We use heatmap to visualize the attention weights as shown in Fig. 3.6. The color scale represents the intensities of attention weights, where a deeper color indicates a higher value and a lighter color indicates a lower value. Fig. 3.6a visualizes the attention weights of a user for 64 items that were purchased by the user (in the *Office* dataset). The x-axis represents the dimensions of attention vector (i.e., 64), and y-axis represents the 64 items. At the beginning, the user attention on all aspects (or dimensions) is set to be the same (i.e., the all the values in the attention vector is 1). During the training process, the attention weight of each dimension will gradually adjust within the range of $[0, 64]$, and finally converge to a constant value. Fig. 3.6a visualizes the converged values, which demonstrates that the user preference on different aspects is varying across different items. Fig. 3.6b illustrates the attention weights of 64 users on the aspects of a randomly selected item (also from the *Office*



(a) user-items



(b) item-users

Fig. 3.6: Visualization of Attention Vector.

dataset). The x-axis also represents the dimensions of attention vector, and y-axis represents different users. We can see that, different users pay attention to different aspects of the same item.

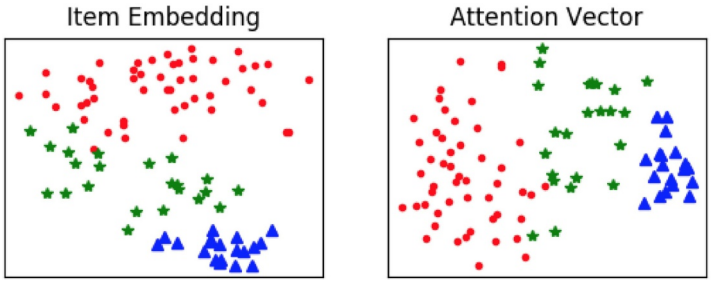


Fig. 3.7: User preference visualization based on item embedding and attention vectors in the Office dataset.

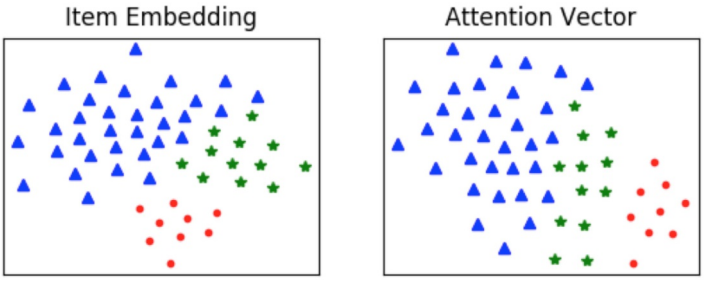


Fig. 3.8: User preference visualization based on item embedding and attention vectors in the WomenClothing dataset.

User diverse preference visualization

We argued that a user preference is diverse, because a user 1) likes items with diverse features; and 2) has different preferences on the aspects of various items. In Fig. 3.7 and Fig. 3.8, we visualize the user preference by clustering the purchased items using 1) item embedding and 2) user attention vectors with respect to items. Fig. 3.7 and Fig. 3.8 show the clustering results of all the items that a user purchased from *Office* and *WomenClothing*, respectively. The left figures visualize the clustering results based on item embedding (i.e., $\boldsymbol{q}_i$); and the right ones illustrate the clustering results based on user attention vectors. Every dot in the figures denotes an item. Besides, in both Fig. 3.7 and Fig. 3.8, the dots with the same color (e.g., blue) in the two figures denote the same set of items. We use t-SNE [120] for clustering and visualization. Notice that the item vector (i.e., $\boldsymbol{q}_i$) is learned based on its interactions with all the users. Therefore, this vector should well characterize all the aspects of the item. In contrast, an attention vector $\boldsymbol{a}_{u,i}$ characterizes a user's attention to the different aspects of an item. Therefore, the clustering results based on the item embedding (left figures) can demonstrate the user diverse preferences on items with different features; the clustering results based on the attention vector (right figures) validate user varying attention on different aspects of items. Besides, we can see

that the distribution of the same items (e.g., red dots in Fig. 3.7) is different in the two types of figures. It also demonstrates that a user pays different attention to the various aspects of different items.

3.6 Conclusion

In this chapter, we presented a novel Multimodal Attentive Metric Learning (MAML) model for top-n recommendation. In particular, this model is designed to model user diverse preferences on different aspects of items. This is achieved by a proposed attention neural network, which exploits the multimodal features of the target item to estimate a user's specific attention on each aspect of this item. Moreover, our model is developed based on a metric-based learning approach, which avoids the inherent limitation of matrix factorization-based methods and thus can capture fine-grained user preferences. Extensive experiments have been conducted on four real-world datasets from Amazon. The results show that our method significantly outperforms a variety of competitors, demonstrating the effectiveness of our method and the potential of modeling user diverse preferences for recommendation. Besides, we also studied the effectiveness of the designed attention mechanism and visualized the user's diverse preferences based on both the learned item vectors and the attention vectors.

Chapter 4
Disentangled Multimodal Representation Learning for Recommendation

4.1 Introduction

As introduced in Chapter 3, it is well-known that users preferences on items are diverse. That is, the intents of a user for liking items are varied across different items. For example, a user may favor a product due to its *quality* and *brand*, and prefer another one just because of its *good-looking appearance*. Because multimodal information provides rich features that directly describe different aspects and even evidences of users' opinion on those aspects (e.g., comments), it has also been exploited to model the diverse preferences of users towards various items. To achieve the goal, existing methods either learn an attention vector for different aspects with respect to the target user-item pair [211] or update the target item/user vector based on the multimodal features of the target items [101, 27, 30]. The success of the aforementioned methods is largely attributed to their effectiveness of exploiting multimodal information to learn better user and item representations. However, we argue that, in addition to modeling users' diverse preferences for different items, it is also important to model users' modality preferences, that is, to mine the contributions of different modality features to users' preference for each factor of an item. The reasons for modeling users' modality preferences are as follows:

Firstly, the prominent factors exhibited by different modalities are different. For example, user reviews contain more user's opinions on the factors of such as *quality* and *comfort* for items [27, 30]. In contrast, item images can help capture more user preferences on the visual appearance (like *color* and *type*) [122, 54]. Existing multimodal-based CF models usually exploit the features of multimodal information without distinguishing the feature differences between different modalities [204, 211]. Apparently, those studies have not considered the different contributions of multimodal information on the factor level for users' preference modeling.

In addition, users have individual differences in perceiving different modalities for organizing information as pointed out in psychology studies [34]. Accordingly, the features of different modalities are of different importance to a user's preference on a certain factor of a target item. In other words, users have different preferences

F. Liu et al., *Multimodal Learning toward Recommendation*,
https://doi.org/10.1007/978-3-031-83188-1_4

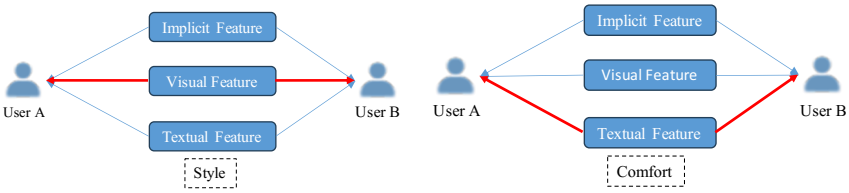


Fig. 4.1: An Example Illustrating Modality Preference. This example demonstrates how different modalities contribute to user preferences. The red line signifies a stronger influence. For the "Style" of an item, visual features play a dominant role in modeling user preferences, whereas for the "Comfort" of an item, textual features are more impactful.

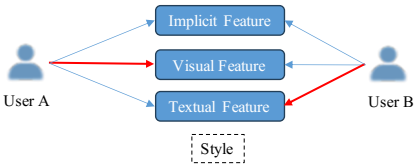


Fig. 4.2: Users have different modality preferences on different items of different factors. For the "Style" of an item, User A prefers visual features (receiving more information from visuals compared to other modalities), while User B prefers textual features. The red line highlights the feature that users pay more attention to.

for different modalities (or *modality preference*) in each factor of an item. For example, the *style* information can be directly viewed from the images or obtained from the descriptions in reviews when purchasing *clothes*. For this factor, some users directly view the images to check whether they like them, while others might be more willing to refer to the opinions in reviews. However, existing multimodal recommendation methods have not considered the modality preferences of users in the utilization of information when exploiting different modalities, resulting in sub-optimal performance.

Finally, the entangled representations of different factors in each modality will also hurt the performance [59, 117, 174]. For example, a user purchases a dress because of its *style* while caring less about *comfort*; otherwise, the user purchases the other dress considering both *style* and *comfort*. When the representations of different factors in each modality are entangled, this will cause information redundancy and performance degradation in preference modeling. We would like to model user preference from different factors and each factor can be represented by the features of different modalities. Disentangled representation has been demonstrated to be more resilient to complex features [59, 117]. Recently, researchers have also applied the disentangled representation learning techniques to model users' diverse intents [174]. Although some progress has been achieved, they only use the user-item interaction data and leave the rich information contained in multimodal data unexploited. In

fact, the multimodal data provides a lot of useful information to help us discover users' intents on items. The problem is how to effectively disentangle and fuse the multimodal features to better serve our recommendation model.

Motivated by the above considerations, in this chapter, we propose a *Disentangled Multimodal Representation Learning* (DMRL for short) model, which models user's preference by considering the different contributions of features from different modalities for each disentangled factor. In our model, a user's preference for an item is predicted by weightedly aggregating her preferences on all modality features of different factors. To yield robust and independent representations for each factor, we adopt a disentangled representation technique to ensure that the features of different factors in each modality are independent of each other. Based on the disentangled representations of different factors, we design a multimodal attention mechanism to capture users' modality preferences on different modalities for each factor. Finally, for the preference prediction, given a user and a target item, we first estimate the user's preference score for each factor in each modality, and then linearly combine the scores of all the factors across different modalities using the estimated weights obtained by the attention mechanism. To this end, our modal can profile users' personal preferences based on disentangled factors represented by multimodalities with the consideration of their personal modality preference. Extensive experiments on five real-world datasets have been carried out to demonstrate the effectiveness of our method with comparison to several strong baselines, including the ones using multimodal information and disentangled learning techniques.

In summary, the main contributions of this work are as follows:

- We highlight the differing importance of multimodal features to various factors of items and claim that users have different modality preferences on different modalities for each factor. Inspired by this, we propose a novel DMRL recommendation method, which models users' preferences by considering the different contributions of multimodal features to each disentangled factor.
- In the model design, we adopt the disentangled representation learning technique to disentangle the factor representations in each modality and design a multimodal attention mechanism to estimate users' attention to the features of different modalities for each factor.
- We have conducted extensive experiments on five real-world datasets to evaluate the effectiveness of our model and perform comprehensive ablation studies to verify the key assumptions of our model. Experimental results show the superiority of our method. In addition, our codes are released to facilitate the replication of our experiments[1].

The rest of this chapter is organized as follows: Section 4.2 provides a comprehensive review of related work. Section 4.3 introduces the preliminaries relevant to our proposed model. Section 4.4 details the components of the proposed model. Experimental results are presented in Section 4.5. Finally, Section 4.6 concludes the chapter.

[1] https://github.com/liufancs/DMRL

4.2 Related Work

In this section, we briefly review the recent advances in multimodal representation in recommendation and disentangled representation learning, which are closely relevant to our work.

4.2.1 Multimodal Representation Learning in Recommendation

Model-based CF models [83, 57, 173, 56, 102] learn the representations of users and items merely relying on the interaction data. To enhance the user and item representations, various types of side information have been utilized in recommender systems, such as attributes [24, 21], reviews [121, 155, 27, 30] and images [122, 168]. Most existing models exploit different types of information individually, because of the challenges of fusing multimodal information. However, it has been well-recognized that features from different modalities are complementary to each other, and the use of multi-modal features can further improve the performance in general [204, 211, 101, 185, 124, 13].

In recent years, researchers have also exploited the multimodal information for learning more comprehensive user and item representation in recommendation systems [204, 211, 63, 101, 182, 185, 36]. Based on how multimodal information is utilised in user and item embedding learning, we broadly classify the existing methods into two types. The first one is to integrate the features learned from different modalities via a linear [204, 182, 185] or non-linear method [211, 101] to obtain a joint representation for each user or item. For instance, JRL [211] concatenates the representations learned from ratings, reviews and images to form the final representation. The other type often adopts a regularization scheme to constrain the relations between representations of different modalities in a latent space [63, 101]. For example, CML [63] uses the multimodal features as a regularization term with a designed loss to facilitate the embedding learning of items and users. For capturing fine-grained user preference, Liu et al. [101] proposed a metric learning-based method, which models diverse user preferences by exploiting the item's multimodal feature. In recent years, Graph Convolution Networks (GCNs) have attracted increasing attention in multimodal recommendation due to the powerful capability on representation learning from non-Euclidean structure [182, 185, 206]. Based on the GCN structure, MMGCN [186] constructs a user-item bipartite graph to learn representations of each modality and then fuse the modality representation together as the final representation. GRCN [185] utilizes the rich multimodal content of items to refine the structure of the interaction graph in order to mitigate the effect of false-positive edges on recommendation performance.

Despite the progress, the factors behind multimodal features are entangled in representations, resulting in sub-optimal performance. In this chapter, in order to yield robust multimodal representations, we employ the disentangled representation technique to encourage the independence of different features.

4.2.2 Disentangled Representation Learning

Disentangled representation learning has gained considerable attention, in particular in the field of image representation learning [59]. It aims to identify and disentangle the underlying explanatory factors behind the data. Such representations have demonstrated to be more resilient to the complex variants [117]. For instance, the beta-VAE method [59] employs a constrained variational framework to learn disentangled representations of fundamental visual concepts. Meanwhile, IPGDN [115] automatically uncovers independent latent factors present in graph data.

In recent years, modeling users' diverse preferences for items has attracted increasing attention in the recommendation field. Cheng et al. [27] applied the topic model on reviews to analyze user preferences on different aspects of items, which are used to model a user's diverse intents to different items. Liu et al. [101] presented a metric-learning based recommendation model, which employs an attentive neural network to estimate user attention on different aspects of the target item by exploiting the item's multimodal features (e.g., review and image). In fact, the factors behind features in the aforementioned methods are highly entangled, which results in sub-optimal performance of recommendation. For learning robust and independent representations from user-item interaction data, disentangled representation begins to be valued in recommendation [118, 176, 174, 89]. Following the previous work in computer vision, the initial attempts take Variational Auto-encoder (VAE) [78] to learn disentangled representations. For example, MacridVAE [118] captured user preferences regarding the different concepts associated with user intentions separately. Beyond the disentangled representations derived from user-item interaction modeling, ADDVAE [159] acquires an additional set of disentangled user representations from textual content and subsequently aligns both sets of representations. For studying the diversity of user intents on adopting the items, DGCF [174] employs a graph disentangled module to iteratively refine the intent-aware interaction graph and factorial representations for recommendations. As the learned disentangled representations lack clear meanings, KDR [126] uses knowledge graphs (KGs) to guide the learning process, ensuring that the resulting disentangled representations are associated with semantically meaningful information extracted from the KGs.

The above methods apply the disentangled learning techniques on interaction data to model users' diverse intents on adopting items without exploiting any side information. We claim that side information, especially the combination of multimodal information (e.g., reviews and images), contain rich evidence about users' preferences on items. In addition, we claim that for a specific factor, the features of different modalities convey different information. Inspired by this consideration, in this paper, we apply the disentangled learning technique on multimodal information to learn a users' preference on each factor of an item. Moreover, we design a multimodal attention network to capture a user's modality preferences on different modalities for each factor.

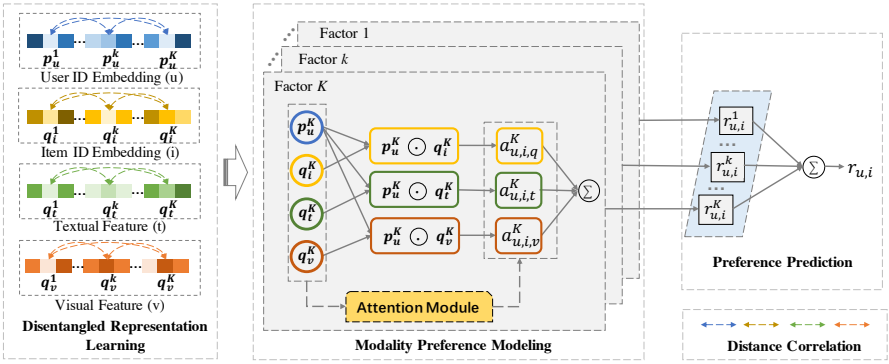


Fig. 4.3: Overview of our DMRL model. Specifically, the disentangled representation technique is adopted to ensure the features of different factors in each modality are independent to each other. A multimodal attention mechanism is then designed to capture the user's modality preference for each factor. Based on the estimated weights obtained by the attention mechanism, DMRL makes recommendations by combining the preference scores of a user's preferences to each factor of the target item over different modalities. Best viewed in color.

4.3 Preliminaries

Before describing our model, we would like to introduce the problem setting first. Given a user set $\mathcal{U}$ and an item set $\mathcal{I}$, we use $\boldsymbol{R}^{N_u \times N_i}$ to denote the user-item interaction matrix, in which a nonzero entry $r_{u,i} \in \boldsymbol{R}$ indicates that user $u \in \mathcal{U}$ has interacted with item $i \in \mathcal{I}$ before; otherwise, the entry is zero. Notice that the interactions can be implicit (e.g., click) or explicit (e.g., rating). N_u and N_i are the numbers of users and items, respectively. In our setting, each user and each item is assigned a unique ID, and it is represented by a trait vector. Besides, each item is associated with multi-modal information, e.g., reviews and images. And for each item, its associated review and image are represented as a textual feature vector and a visual feature vector, respectively. It should be noted that item ID, image and review are regarded as three types of modality in our setting. Our goal is to recommend a user $u \in \mathcal{U}$ with suitable items which the user did not purchase before and will find appealing.

Existing multimodal-based methods often represent each modality (i.e., ID, review, and image) of each item as a holistic vector. As aforementioned, we would like to model users' diverse preference on different factors, and more specifically, their attention to different modalities of the same factor. To achieve the goal, in our model, the vector of each modality is composed of K chunks, with the assumption that each chunk represents a factor (such as appearance or quality). Without loss of generality, we use the user ID embedding as an example, it is initialized as:

$$\boldsymbol{p_u} = (\boldsymbol{p_u^1}, \boldsymbol{p_u^2}, \cdots, \boldsymbol{p_u^K}), \tag{4.1}$$

where $\boldsymbol{p_u} \in \mathbb{R}^d$ is ID embedding to capture intrinsic characteristics of user u; $\boldsymbol{p_u^k} \in \mathbb{R}^{\frac{d}{k}}$ is u's chunked representation of the k-th factor. Analogously, $\boldsymbol{q_i} = (\boldsymbol{q_i^1}, \boldsymbol{q_i^2}, \cdots, \boldsymbol{q_i^K})$, $\boldsymbol{q_t} = (\boldsymbol{q_t^1}, \boldsymbol{q_t^2}, \cdots, \boldsymbol{q_t^K})$ and $\boldsymbol{q_v} = (\boldsymbol{q_v^1}, \boldsymbol{q_v^2}, \cdots, \boldsymbol{q_v^K})$ are defined as the embeddings of item i's ID, textual feature and visual feature, respectively.

The embeddings of user ID and item ID are randomly initialized in our model, and the initial embeddings of textual and visual features of items are extracted from reviews and images, which are described in the following.

Feature Extraction

In our implementation, the BERT [35] model is adopted to extract the text feature. Specifically, BERT learns continuous distributed vector representations for textual documents in an unsupervised way. BERT tries to preserve the semantic information and also considers the sequential information in the word sequence. It takes text documents as inputs and outputs their vector representations in a latent semantic space. The visual feature $\boldsymbol{F}_v$ is extracted by the ViT [81] model that adopts transformer encoder with 32 layers. In this work, we take the output of the transformer encoder, resulting in a 1024-D feature vector as the visual features for each item[2].

The extracted textual and visual features are then fed into two separate two-layer neural networks, which are used as refinement networks to learn valuable features oriented for recommendation [3]. Specifically, the two-layer neural network is:

$$\boldsymbol{q}_x = \sigma(\boldsymbol{W}_x^0 \sigma(\boldsymbol{W}_x^1 \boldsymbol{F}_x + \boldsymbol{b}_x^1) + \boldsymbol{b}_x^0), \tag{4.2}$$

where $\boldsymbol{F_x}$ is the input feature vector of the x-modality (i.e., visual or textual). $\boldsymbol{W}_x^0$, $\boldsymbol{W}_x^1$ and $\boldsymbol{b}_x^0$, $\boldsymbol{b}_x^1$ are the corresponding weight matrices and bias vectors for two layers, respectively. $\sigma(\cdot)$ is the activation function and LeakyReLU is adopted because its biologically plausible and non-saturated property [173]. The textual feature $\boldsymbol{F_t}$ and visual feature $\boldsymbol{F_v}$ are processed by two separate networks to obtain the embeddings of $\boldsymbol{q_t}$ and $\boldsymbol{q_v}$ respectively for subsequent learning. Noted that our focus in this chapter is to study whether the disentangled representation of factors can help us enhance the recommendation performance. The above textual and visual extraction methods can be replaced by any other more advanced methods. In the next, we give the details of our disentangled multimodal representation learning recommendation model.

4.4 Our Model

As shown in Figure 4.3, our *Disentangled Multimodal Representation Learning* (DMRL) consists of three key components: 1) **disentangled representation learning**, which employs distance correlation [152, 151] as a regularizer to encourage the independence of the features of different factors in each modality; 2) **modality**

[2] In this chapter, each item is associated with one image.

[3] Experiments show that the neural network with two layers could achieve better performance.

preference modeling, which captures user modality preference on different modalities for each factor; 3) **preference prediction**, which predicts the preference of a user to the target item by aggregating the learned disentangled features from different modalities with assigned attention weights. In the next, we introduce each component in sequence.

4.4.1 Disentangled Representation Learning

We would like to divide the feature representations of each modality into k chunks. And each chunk represents a latent factor which impacts user's preference on the item [4]. For simplicity, the features from each modality are evenly divided into k continuous chunks. As a result, the features of different factors will be entangled. This will cause information redundancy and reduce the efficacy of preference modeling based on those factors. To avoid the problem, we need to first disentangle the features of different factors in each modality. To achieve the goal, we apply distance correlation [151, 152] to characterize independence of any two-paired embeddings of different factors. Its coefficient is zero if and only if these vectors are independent. For a feature vector y, we formulate it as:

$$L_y = \sum_{k=1}^{K} \sum_{k'=k+1}^{K} dCor\left(\boldsymbol{y}^k, \boldsymbol{y}^{k'}\right), \tag{4.3}$$

where $dCor(\cdot)$ is the function of distance correlation and it is defined as:

$$dCor\left(\boldsymbol{y}^{\boldsymbol{k}}, \boldsymbol{y}^{\boldsymbol{k}'}\right) = \frac{dCov\left(\boldsymbol{y}^{\boldsymbol{k}}, \boldsymbol{y}^{\boldsymbol{k}'}\right)}{\sqrt{dVar\left(\boldsymbol{y}^{\boldsymbol{k}}\right) dVar\left(\boldsymbol{y}^{\boldsymbol{k}'}\right)}}, \tag{4.4}$$

where $\boldsymbol{y}^{\boldsymbol{k}}$ and $\boldsymbol{y}^{\boldsymbol{k}'}$ are respectively the k and k'-th chunks of the vector y; $dCov(\cdot)$ represents the distance covariance between two matrices; $dVar(\cdot)$ is the distance variance of each matrix. In this way, we can define the regularization losses L_p, L_q, L_t, L_v to encourage the feature independence of the factors in user ID, item ID, textual feature and visual feature, respectively.

Finally, the regularization loss for disentangled learning is formulated as:

$$L_d = L_p + L_q + L_v + L_t. \tag{4.5}$$

Note that in this equation, the weights of different regularization losses (L_p, L_q, L_v, L_t) can be empirically tuned. Here, we set them to be equal in the learning process for simplicity.

[4] Note that we use the semantic factors, such as *appearance* and *quality*, as examples for ease understanding of our model. Latent factors are implicit and unexplainable.

4.4.2 Modality Preference Modeling

Typically, for a certain factor, users often value its features exhibited by different modalities differently. For example, for the *appearance* of a product, a user will take the visual features (from images) more seriously than the textual features (from reviews); and value the textual features more than the visual features for the *quality* of the product. To capture the complicated user modality preferences on various modalities for each factor, we design a weight-shared multimodal attention mechanism in DMRL. Specifically, for an item i, a user u's specific attention $a^k_{u,i}$ to different modalities for the k-th factor is computed as:

$$\boldsymbol{h^k_{u,i}} = Tanh(\boldsymbol{W}[\boldsymbol{p}^k_u; \boldsymbol{q}^k_i; \boldsymbol{q}^k_t; \boldsymbol{q}^k_v] + \boldsymbol{b}), \tag{4.6}$$

$$\boldsymbol{\hat{a}^k_{u,i}} = \mathbf{W}_v \boldsymbol{h^k_{u,i}}, \tag{4.7}$$

where $\boldsymbol{W}$ and $\boldsymbol{b}$ are respectively the weight matrix and bias vector of the neural network. In order to enhance efficiency of learning, we use shared weight under all factors. $\boldsymbol{h^k_{u,i}}$ is an output of of the hidden layer; $\mathbf{W}_v$ is a transformation matrix that projects the hidden layer into an output attention weight vector. $[\boldsymbol{p}^k_u; \boldsymbol{q}^k_i; \boldsymbol{q}^k_t; \boldsymbol{q}^k_v]$ denotes the concatenation of $\boldsymbol{p}^k_u$, $\boldsymbol{q}^k_i$, $\boldsymbol{q}^k_t$ and $\boldsymbol{q}^k_v$. *Tanh* is used as the activation function.

Following the standard procedure of attention mechanism, there is a subsequent step to normalize $\boldsymbol{\hat{a}^k_{u,i}}$ with the softmax function, which converts the attention weights to a probability distribution. In our model, the final attention weight for the textual feature of the k-th factor is computed as:

$$a^k_{u,i,t} = \frac{exp\left(\hat{a}^k_{u,i,t}\right)}{exp\left(\hat{a}^k_{u,i,q}\right) + exp\left(\hat{a}^k_{u,i,t}\right) + exp\left(\hat{a}^k_{u,i,v}\right)}. \tag{4.8}$$

Similarly, the weight for item i's ID and visual feature $a^k_{u,i,q}$ and $a^k_{u,i,v}$ can be computed in the same way.

4.4.3 Preference Prediction

User's preference for an item should be estimated based on her preferences on different modalities of all factors for this item. Hence, with the representations of different modalities for different factors, given a user u and a target item i with their representations, the user preference on the k-th factor according to the features of the x-modality is estimated as:

$$r^k_{u,i,x} = a^k_{u,i,x} \cdot \sigma(\boldsymbol{p^k_u} \odot \boldsymbol{q^k_x}). \tag{4.9}$$

where $\odot$ denotes dot product. Notice that the attention weight $a^k_{u,i,x}$ represents the importance of the x-modality to the k-th factor for the user's preference to the item; and the dot product between $\boldsymbol{p}^{\boldsymbol{k}}_{\boldsymbol{u}}$ and $\boldsymbol{q}^{\boldsymbol{k}}_{\boldsymbol{x}}$ estimates how the features of the x-modality fits this user's tastes on the k-th factor. The integration of both terms can evaluate the user's preference to the item on the k-th factor according to the features of x-modality more comprehensively. $\sigma(\cdot)$ is the activation function and softplus is adopted to ensure that the yielded score is positive. Similarly, we can compute $r^k_{u,i,q}$, $r^k_{u,i,t}$ and $r^k_{u,i,v}$ in the same way, which represent the user preferences on the k-th factor from different modalities, i.e., item ID, reviews, and images, respectively. And the final score of user preference on the k-th factor of the target item is predicted by aggregating the scores from different modalities:

$$r^k_{u,i} = r^k_{u,i,q} + r^k_{u,i,t} + r^k_{u,i,v}. \tag{4.10}$$

Finally, the predicted scores of all the factors are integrated together to predict user preference to the target item, which is:

$$r_{u,i} = \sum_{k=1}^{K} r^k_{u,i}. \tag{4.11}$$

4.4.4 Model Learning

Objective function

We target at the top-n recommendation, namely, to recommend a set of n top-ranked items which match the target user's preferences. Similar to other rank-oriented recommendation studies [211, 173], we use the pairwise-based learning method for optimization. The loss function is defined as:

$$L_{BPR} = \sum_{(\mathbf{u},\mathbf{i}^+,\mathbf{i}^-)\in O} -\ln \phi(r_{u,i} - r_{u,i^-}), \tag{4.12}$$

where $O = \{(u, i^+, i^-)|(u, i^+) \in \mathcal{R}^+, (u, i^-) \in \mathcal{R}^-\}$ denotes the training set; $\mathcal{R}^+$ indicates the observed interactions between user u and i^+ in the training dataset, and $\mathcal{R}^-$ is the sampled unobserved interaction set. For each positive pair (u, i^+), we first randomly sample n negative ones from the items that a user hasn't purchased before as candidate items.[5] From the candidate items, we select the one which is most similar to the user u based on the dot product of their embeddings, as a hard negative sample to construct the negative pair (u, i^-).

[5] In our implementation, we empirically set $n = 4$ to balance efficiency and effectiveness.

Optimization

With the consideration of all the regularization terms, the final object function of our DMRL is,

$$loss = L_{BPR} + \lambda_\theta L_\theta + \lambda_d L_d, \tag{4.13}$$

where $L_\theta = \|\Theta\|_2^2$ represents the L_2 regularization for the parameters Θ of the model; λ_θ and λ_d are hyperparameters that control the weights of L_2 regularizer and independence regularizer, respectively. The optimization is quite standard and the stochastic gradient descent (SGD) algorithm is adopted. In implementation, the Adam optimizer [77] is adopted to tune the learning rate.

4.5 Experiments

To validate the effectiveness of our model, we conducted extensive experiments on five public datasets to answer the following research questions:

RQ1: Does the proposed MAML enhance the performance of state-of-the-art recommendation models for top-n recommendation tasks?

RQ2: How does modality preference influence the performance of multimodal recommendation models?

RQ3: Can our method effectively disentangle multimodal representations for the recommendation?

RQ4: What is the contribution of each component in DMRL?

RQ5: How does the number of factors affect the performance of DMRL?

In the next, we first introduce the experimental setup, and then report and analyze the experimental results.

4.5.1 Experimental Setup

Datasets

The public Amazon review dataset[6] [121], which has been widely used for recommendation evaluation in previous studies, is used for evaluation in our experiments. This dataset contains user interactions (review, rating, helpfulness votes, etc.) on items as well as the item metadata (descriptions, price, brand, image features, etc.) on 24 product categories. Five product categories in this dataset are selected and all the reviews and item images are kept as side information for items. We pre-processed

[6] http://jmcauley.ucsd.edu/data/amazon.

Table 4.1: Basic statistics of the experimental datasets.

Dataset	#user	#item	#interactions	sparsity
Office	4,874	7,279	52,957	99.85%
Baby	12,637	18,646	121,651	99.95%
Clothing	18,209	35,526	150,889	99.98%
Toys Games	18,748	30,420	161,653	99.97%
Sports	21,400	36,224	982,618	99.97%

the dataset to only keep the items and users with at least 5 interactions. The basic statistics of the five datasets are shown in Table 4.1. For each observed user-item interaction, we treated it as a positive instance, and then paired it with negative items which are randomly sampled from items that the user has not purchased before.

In this work, we focus on the top-*n* recommendation task, which aims to recommend a set of top-*n* ranked items that will be appealing to the target user. For each dataset, we randomly selected 80% of the interactions from each user to construct the training set, and the remaining 20% for testing. From the training set, we randomly selected 10% of interactions as a validation set to tune hyper-parameters.

Baselines

We compare our DMRL model with the following baselines, including both deep (NeuMF [57], JRL [211]) MF and metric learning (CML [63], MAML [101]) based models, as well as the graph-based methods (NGCF [173], DGCF [174], MMGCN [182], GRCN [185]). In experiments, for all the baselines, we use the codes they released for evaluation. And, we put great effort to tune hyperparameters of these methods and reported their best performance.

Evaluation Metrics

For each user in the test set, we treat all the items that the user did not interact with as negative items. Two widely used metrics for top-*n* recommendation are adopted in our evaluation: *Recall* and *Normalized Discounted Cumulative Gain* (NDCG) [55]. For each metric, the performance is computed based on the top 20 results. Notice that the reported results are the average values across all the testing users.

Parameter Settings

We implemented our model with Tensorflow [7] and carefully tuned the key parameters. The embedding size is fixed to 128 for all methods and the embedding parameters are initialized with the Xavier method [190]. We optimized our method

[7] https://www.tensorflow.org.

Table 4.2: Performance of our DMRL model and the competitors over *Office*, *Baby* and *Clothing*. The best results are highlighted in bold. We categorize the methods into two categories (i.e., blocks): 1) methods in the 1st block are the ones relying only on the user-item interaction information without any side information, and 2) methods in the 2nd block are the ones that consider both text and image features.

Datasets	Office		Baby		Clothing	
Metrics	Recall	NDCG	Recall	NDCG	Recall	NDCG
NeuMF	0.0605	0.0337	0.0502	0.0224	0.0189	0.0080
CML	0.1229	0.1395	0.0674	0.0444	0.0409	0.0224
NGCF	0.0814	0.0479	0.0694	0.0313	0.0466	0.0216
DGCF	0.1206	0.1105	0.0788	0.0465	0.0745	0.0405
JRL	0.0656	0.0365	0.0579	0.0266	0.0233	0.0124
MMGCN	0.1173	0.1231	0.0814	0.0496	0.0573	0.0312
MAML	0.1333	0.1412	0.0867	0.0521	0.0782	0.0387
GRCN	0.1351	0.1524	0.0883	0.0541	0.0861	0.0452
DMRL	**0.1563***	**0.1842***	**0.0906***	**0.0561***	**0.0917***	**0.0511***

The symbol * denotes that the improvement is significant with $p - value < 0.05$ based on a two-tailed paired t-test.

with Adam [77] and used the default learning rate of 0.0001 and default mini-batch size of 1024. The L_2 regularization coefficient λ_θ and the distance correlation λ_d are searched in the range of $\{1e^{-5}, 1e^{-4}, \cdots, 1e^{+1}\}$. We search the number of the factors being disentangled in $\{1, 2, 4, 8\}$. Besides, model parameters are saved in every 10 epochs. The early stopping strategy [173] is performed, $i.e.$, premature stopping if recall@20 does not increase for 50 successive epochs. For fair comparisons, for the methods using the negative sample strategy, we paired each positive instance in the training set with four randomly sampled negative instances to train them.

4.5.2 Performance comparison (RQ1)

The results of our model and all the competitors over the five datasets are reported in Table 4.2 and 4.3. Overall, it can be seen that our method outperforms all the competitors consistently across all the datasets in terms of different metrics. By grouping all the methods into two categories based on whether using multimodal information, we gain some interesting observations.

The methods in the first block only use the user-item interactions. NeuMF models the non-linear interactions between users and items by using deep neural networks and achieves better performance than the traditional MF methods using linear interactions [57]. NGCF exploits high-order connectivities between users and items through embedding propagation over the graph structure. Hence it obtains better

Table 4.3: Performance of our DMRL model and the competitors over *Toys Games* and *Sports*. The best results are highlighted in bold. We categorize the methods into two categories (i.e., blocks): 1) methods in the 1st block are the ones relying only on the user-item interaction information without any side information, and 2) methods in the 2nd block are the ones that consider both text and image features.

Datasets	Toys Games		Sports	
Metrics	Recall	NDCG	Recall	NDCG
NeuMF	0.0253	0.0128	0.0330	0.0157
CML	0.1227	0.1160	0.0982	0.0640
NGCF	0.0970	0.0587	0.0707	0.0337
DGCF	0.1262	0.1085	0.1026	0.0629
JRL	0.0472	0.0413	0.0368	0.0214
MMGCN	0.1171	0.1056	0.0913	0.0572
MAML	0.1183	0.1117	0.1029	0.0676
GRCN	0.1336	0.1236	0.1065	0.0693
DMRL	**0.1434***	**0.1331***	**0.1111***	**0.0711***

The symbol * denotes that the improvement is significant with $p - value < 0.05$ based on a two-tailed paired t-test.

user and item representations than NeuMF. DGCF outperforms NGCF across all the datasets. This is attributed to the utilization of disentangled representation to learn robust and independent user and item embeddings by considering users' diverse intents. However, the aforementioned methods all use the dot product as the interaction function. As pointed out in [63], dot product does not obey the triangle inequality and thus cannot well model fine-grained user preferences. By using a metric-based learning approach, CML outperforms both NeuMF and NGCF by a large margin. This observation is also consistent with the results reported in [101].

All the methods in the second block exploit both textual and visual features besides the interaction information. In general, these methods yield better performance than those without using multimodal features, demonstrating the effectiveness of leveraging side information on modeling user preference. Owing to the valuable information in multimodal features, JRL outperforms NeuMF accross all the datasets. By exploiting user-item interactions to guide the representation learning in different modalities, MMGCN yields better performance over NGCF on all the datasets. However, MMGCN underperforms DGCF on four datasets, which is because the entangled multimodal representations limit the capability of representation learning. MAML outperforms MMGCN across all the datasets, owing to the adoption of metric-based learning approach and attention mechanism to capture user diverse preferences. By discovering and pruning potential false-positive edges, GRCN obtains better performance across all datasets.

DMRL outperforms all the baselines consistently over all the datasets. We credit this to the joint effects of the following four aspects. Firstly, the utilization of mul-

Table 4.4: Performance of our DMRL model and its variants over five datasets. The best results are highlighted in bold.

Datasets	Office		Baby		Clothing		ToysGames		Sports	
Metrics	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG	Recall	NDCG
DGCF	0.1206	0.1105	0.0788	0.0465	0.0745	0.0405	0.1262	0.1085	0.1026	0.0629
DMRL$_t$	0.1517	0.1578	0.0783	0.0497	0.0766	0.0419	0.1258	0.1257	0.1071	0.0657
DMRL$_v$	0.1532	0.1612	0.0852	0.0534	0.0897	0.0464	0.1395	0.1287	**0.1122**	0.0682
DMRL$_{w/o\ a}$	0.1513	0.1454	0.0759	0.0479	0.0781	0.0392	0.1363	0.1262	0.0998	0.0597
DMRL$_{w/o\ u}$	0.1541	0.1559	0.0835	0.0513	0.0841	0.0427	0.1211	0.1186	0.1112	0.0691
DMRL	**0.1563**	**0.1842**	**0.0906**	**0.0561**	**0.0917**	**0.0511**	**0.1434**	**0.1331**	0.1111	**0.0711**

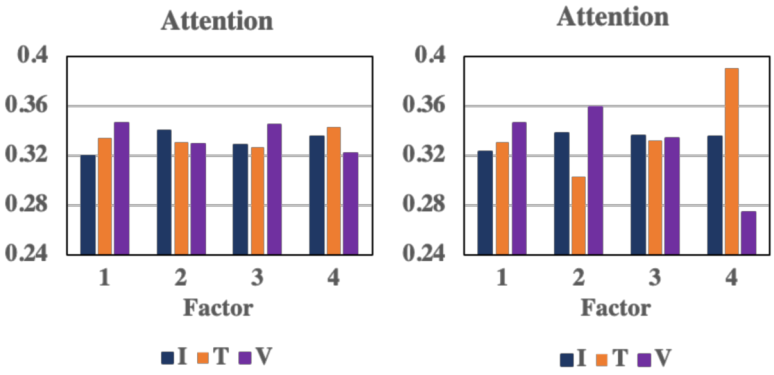
timodal information on modeling user preference, which can be observed from the performance comparisons of methods between two blocks in Table 4.4. Secondly, DMRL models user preference by considering the different contributions of different modalities to each factor. Thirdly, an attention network is designed in our model to capture user's modality preference on different modalities for each factor. Finally, the use of the disentangled representation learning technique is adopted to learn independent representations for different factors. As demonstrated in DGCF, disentangled representation can better model users' multiple intents and thus achieve better performance.

4.5.3 Effects of Modality Preference (RQ2)

One of our assumptions in the model is that for each factor of an item, different users may value its features of one modality more than those of another modality. To capture this, we design an attention mechanism to compute the attention weights for each factor of different modalities. The other assumption is that different modalities contribute differently for a user's preference to an item. With this consideration, for each user-item pair, our model computes the rating given to each factor of an item across different modalities (see Eq. 4.9). In this section, we would like to validate the above two assumptions via visualization: 1) visualizing the learned attention weights to illustrate user's modality preferences on different modalities for each factor; and 2) visualizing the user preferences (ratings) to illustrate two users' preferences on different factors represented by different modalities.

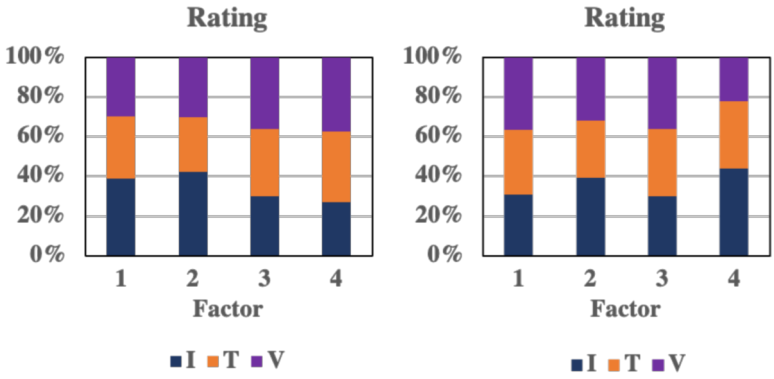
Modality Preferences (Attention Weights)

In order to study the user's modality preferences on different modalities for each factor, we compute the attention weights for different modalities of each factor (in



(a) User A (b) User B

Fig. 4.4: Attention weights of different modalities. I, T and V represent item ID, textual feature and visual feature, respectively.

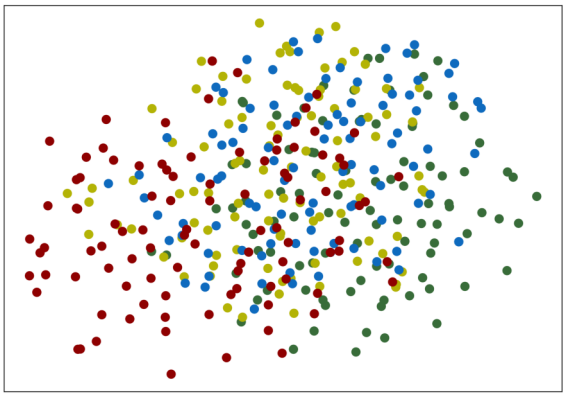


(a) User A (b) User B

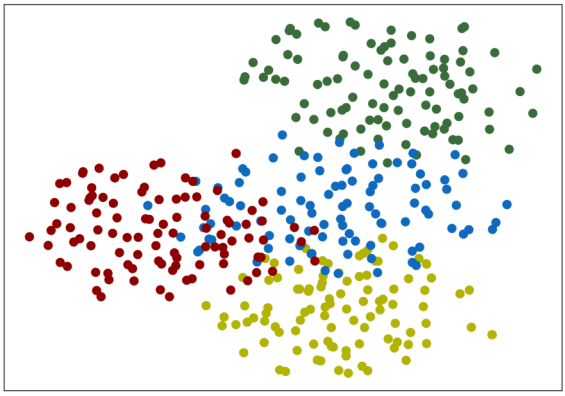
Fig. 4.5: Ratings from two users. I, T and V represent item ID, textual feature and visual feature, respectively.

the *Office* dataset)[8],which are visualized in Fig. 4.4. Specifically, I, T and V represent item ID, textual feature and visual feature, respectively. The x-axis represents different factors. Comparing the results in Fig. 4.4(a) and Fig. 4.4(b), we can see that the weights of different modalities for the same factor are different between the two users. This verifies that for an item, the preferences of users to different modalities are different for each factor. In addition, it should be noted that the attention weight

[8] We selected two users (0 and 197) who purchased the same item (1294) as an example.



(a) Entangled Vector



(b) Disentangled Vector

Fig. 4.6: Visualization of entangled and disentangled vectors extracted from textual features. Best viewed in color.

cannot represent the contribution of a modality to a user's preference to a factor of the target item. Specifically, take the results in the 4-th factor in Fig. 4.5(b) and Fig. 4.4(b) for example, we can find that although the user pays more attention to the textual feature according to the attention weight, the item ID actually contributes more to the user preference.

User Preferences (Ratings)

In order to further study the contributions of different modalities to a user's preference on the factors of the target item, we compute the ratings of the two different

users given to different factors of the same item represented by different modalities. The ratings are then normalized across different modalities and visualized in Fig. 4.5. Overall, both Fig. 4.5(a) and Fig. 4.5(b) show that for the same factor, the features of different modalities have different contributions to user preference. For example, item ID yields a higher score for the second factor than the scores of both visual feature and textual feature. In other words, the textual and visual features contribute relatively less to the user's preference on this factor. What's more, take the results in the 4-th factor in Fig. 4.5(b) and Fig. 4.4(b) for example, we can find that although the user pays more attention to the textual feature according to the attention weight, the item ID actually contributes more to the user preference. Besides, the variations between Fig. 4.5(a) and Fig. 4.5(b) also demonstrate the different modality preferences of users to different modalities.

4.5.4 Effect of Disentanglement (RQ3)

In Fig. 4.6, we visualized the entangled and disentangled vectors extracted from textual features. The textual features is associated with the items from *Office* [9]. In both Fig. 4.6a and Fig. 4.6b, the dots with the same color (e.g., red) denote the vectors of the same factor. We use t-SNE [120] for clustering and visualization. The clustering results based on the vectors of different factors (right figure) can illustrate the effectiveness of using disentangled representation technique. In other words, it demonstrates that our proposed method could encourage independence of the vectors of different factors for modality features.

4.5.5 Ablation Study (RQ4)

In this section, we examine the contribution of different components to the performance of our model. Our analyses are based on the performance comparisons to the following methods.

- **DGCF [174]**: It is a disentangled representation based method which only uses item IDs.
- **DMRL**$_t$: It is a variant of our method which only uses item IDs and textual features.
- **DMRL**$_v$: It is a variant of our method which only uses item IDs and visual features.
- **DMRL**$_{w/o\ a}$: This variant removes the designed multimodal attention mechanism. It exploits the multimodal features of different factors indiscriminately.
- **DMRL**$_{w/o\ u}$: This variant removes user information in Eq. 4.6, namely, for the same item, the attention weights are all the same for different users. It is designed

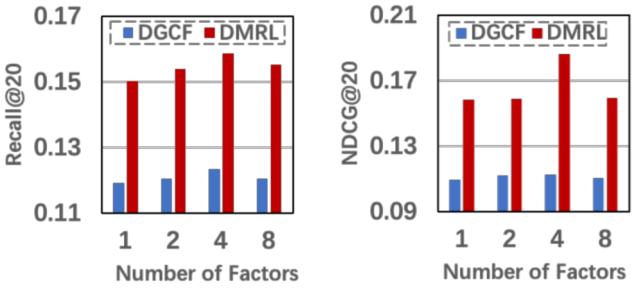
[9] We randomly sampled the items and their associated textual features.

to investigate the effectiveness of modeling user modality preference on different modalities by comparing to our DMRL model.

The results of those variants and our method are reported in Table 4.4. We group all the methods and variants into two groups based on whether the methods use all the modalities. The methods in the first block only use a part of three modalities. Both DMRL$_t$ and DMRL$_v$ outperform DGCF, demonstrating the effectiveness of leveraging side information to learn user preference. DMRL$_v$ outperforms DMRL$_t$ in terms of Recall@20 over all the datasets, which indicates the importance of visual features on user preference modeling. DMRL$_t$ performs better than DMRL$_v$ in terms of NDCG@20 on *ToysGames* and *Sports*. This is because textual features have a higher contribution for the two cases.

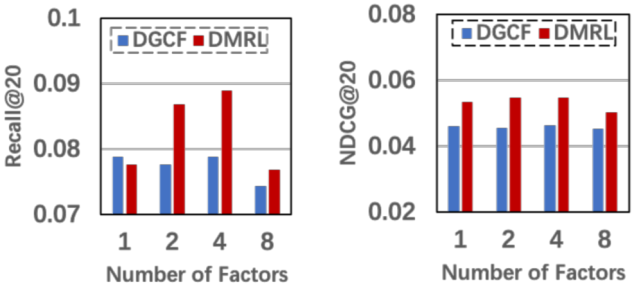
The variants in the second block use features of all three modalities. DMRL$_{w/o\ u}$ outperforms DMRL$_{w/o\ a}$ in almost all the cases, which indicates the importance of distinguishing different contributions of different modalities. More importantly, without the attention mechanism, DMRL$_{w/o\ a}$ even underperforms DMRL$_v$ and DMRL$_t$, which use our attention method. This further demonstrates the importance of modeling users' modality preferences. The further improvement of DMRL over DMRL$_{w/o\ u}$ validate the effectiveness of distinguishing different users' attentions to various modalities for each factor.

It is surprising that the methods in the second block underperform the methods in the first block on *Sports* in terms of Recall@20. This warns us that it is possible that the features of different modalities are inconsistent, which may exert a negative influence on preference modeling. More advanced multimodal feature fusion methods are needed to tackle this problem. Besides this special case, DMRL can achieve superior performance over DMRL$_v$ and DMRL$_t$ consistently, which indicates the benefits of considering multimodal information in recommendation. The comprehensive ablation studies also demonstrate the positive utility of different components of our DMRL model.



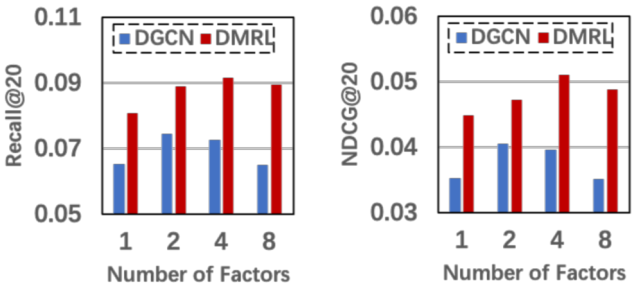
(a) Recall on Office (b) NDCG on Office

Fig. 4.7: Impact of the Factor Number (K) on Office dataset.



(a) Recall on Baby (b) NDCG on Baby

Fig. 4.8: Impact of the Factor Number (K) on Baby dataset.



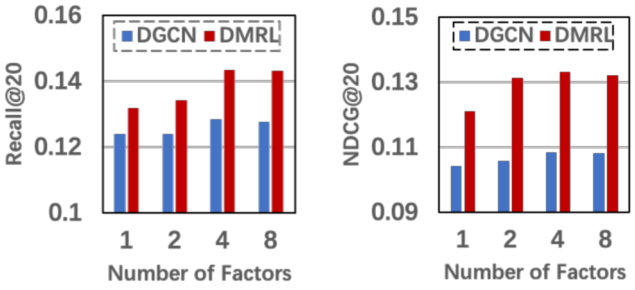
(a) Recall on Clothing (b) NDCG on Clothing

Fig. 4.9: Impact of the Factor Number (K) on Clothing dataset.

4.5.6 Effects of the Factor Number (RQ5)

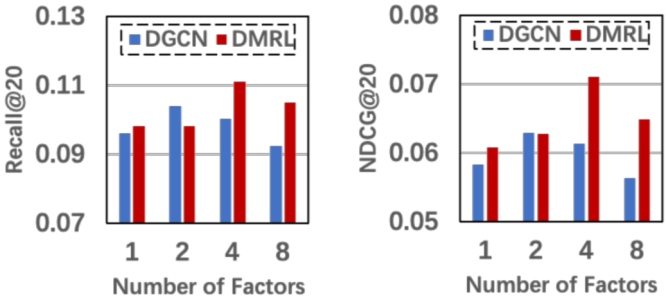
In this section, we study the influence of the factor number on the performance of our method. To compare the performance of our model and the baseline model with different factor numbers, we carry out experiments by tuning the number of factors in $\{1, 2, 4, 8\}$ on different datasets.

We compare our model to DGCF which also uses disentangled representations but did not use multimodal features. Fig. 4.7, 4.8, 4.9, 4.10, 4.11 show the results in terms of Recall@20 and NDCG@20 on all five datasets. As we can see, with the increasing of the number of factors, both DMRL and DGCF achieve better performance and the best performance is obtained when the number is 2 or 4. This indicates that recommendation methods can benefit from robust and independent representation learning technique. Specifically, DGCF performs worse over *Office*, *Baby* and *Clothing* when $K = 1$ and $K = 2$, indicating that an insufficient number



(a) Recall on ToysGames (b) NDCG on ToysGames

Fig. 4.10: Impact of the Factor Number (K) on ToysGames dataset.



(a) Recall on Sports (b) NDCG on Sports

Fig. 4.11: Impact of the Factor Number (K) on Sports dataset.

of factors limits the capability of disentangled representation. However, the recommendation performance drops when when $K = 8$. The reason might be that the chunked representations with small embedding size have limited expressiveness in representation learning for each factor [174]. The large improvement over DGCF shows the effectiveness of our model on exploiting multimodal features.

4.6 Conclusion

In this chapter, we present a novel recommendation model called Disentangled Multimodal Representation Learning (DMRL), which models user's modality preference with respect to multimodal information on different factors of items. In DMRL, we adopt a disentangled representation learning technique to disentangle the represen-

tations of different factors in each modality. In addition, we design a multimodal attention mechanism to capture user's modality preferences to different modalities for each factor, so as to learn better representations for recommendation. Finally, we estimate a user's preference score to each factor of the target item based on its representation in each modality, and then combining the scores together across all modalities and factors with the computed attention weights. Extensive experiments on five publicly available datasets show that our model outperforms the state-of-the-art methods, demonstrating the superiority of our method on exploiting multimodal information. The ablation studies further validate the importance of modeling personal modality preference to different modalities.

Chapter 5
Dynamic Multimodal Fusion via Meta-Learning Towards Multimodal Recommendation

5.1 Introduction

After extracting effective multimodal features of a user or an item in Chapter 2 and subsequently modeling modality preferences in Chapter 3, a crucial next step is to integrate these multimodal features into a joint multimodal representation, known as multimodal fusion, to comprehensively represent the user or item. Existing multimodal fusion methods can be roughly classified into two categories: linear [185] and nonlinear [37] methods. For example, MMGCN [186] performs the graph convolutional operations on user-item graphs in different modalities, and implements multimodal fusion via the linear combination of multimodal representations. Due to the limitations of linear methods, deep neural networks are employed to fuse multimodal information nonlinearly. For instance, VBPR [54] adopts a neural layer to project the multi-modality features into the latent space. ACF [20] introduces a hierarchical attention mechanism to select informative content by assigning weights to different modalities.

Despite the remarkable performance, the previous methods are only able to perform static multimodal fusion as shown in Figure 5.1(a), following the assumptions that: 1) for different items, the information from the same modality plays a similar role and the multimodal information obeys the same relationship; and 2) a fusion function can be learned from the vast amount of items, and that the function learned is capable of modeling the relationships among the multimodal information. However, we argue that the above-mentioned assumptions are fragile in recommendation. Taking the recently popular micro-videos as an example, unlike the well-designed video (*e.g.*, movies or documentaries), the modality-specific information of different micro-videos may have very different effects on the expression of the theme, and the relationships among modalities might also be distinct [6]. For instance, the information in visual and acoustic modalities is complementary in music micro-videos; while in education micro-videos, acoustic and textual modalities convey the themes in a consistent relationship. With this consideration, we argue that static fusion is under-fitting the varying multimodal fusion for learning representations of items,

F. Liu et al., *Multimodal Learning toward Recommendation*,
https://doi.org/10.1007/978-3-031-83188-1_5

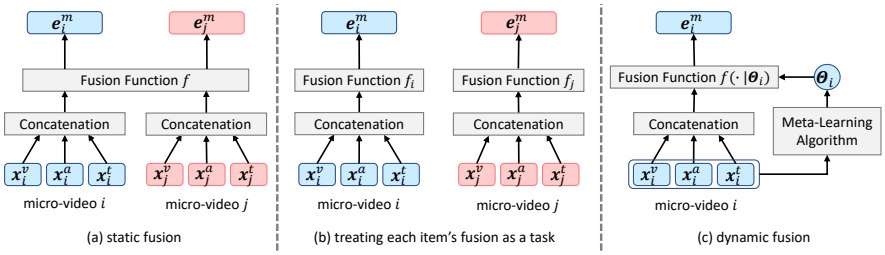


Fig. 5.1: Illustration of the static and the dynamic multimodal fusion, exemplified using micro-videos. $\mathbf{x}_i^v$, $\mathbf{x}_i^a$, and $\mathbf{x}_i^t$ denote the features derived from visual, acoustic, and textual modalities of micro-video i, respectively. $\mathbf{e}_i^m$ denotes the multimodal representation of micro-video i through the fusion function. Θ_i denotes the item-specific parameters dynamically generated for the fusion function by a meta-learning algorithm. Similar notations are used for denoting the attributes of micro-video j.

leading to suboptimal recommendation performance. Furthermore, we consider that each item should combine its visual, acoustic, and textual modality information in a unique way.

In this chapter, using micro-videos as an example, we study the issues involved in multimodal fusion mentioned above. We propose a dynamic multimodal fusion method that treats the multimodal fusion of each item as an independent task. As shown in Figure 5.1(b), the item-specific fusion functions are learned for different items independently. For implementation, we resort to the meta-learning techniques to establish the dynamic multimodal fusion as shown in Figure 5.1(c), due to their following advantages:

- Meta-learning algorithm is capable of generating adaptive parameters according to the context of a single task [221, 41], which is consistent with the target of dynamic multimodal fusion strategy.
- Meta-learning algorithm offers an effective way to perform transfer learning across tasks by means of shared parameters [161], thus enabling us to effectively learn the item-specific fusion functions with the limited training samples.

More specifically, we develop a novel meta-learning-based multimodal fusion model, named Meta Multimodal Fusion (MetaMMF), to dynamically integrate multimodal information for micro-video recommendation. In particular, MetaMMF consists of two main components: a meta information extractor and a meta fusion learner. The former is implemented by a multi-layer perceptron (MLP). It projects the multimodal features of the input task (for an item) into the meta information that represents the task features in the form of higher-order abstraction [128]. Based on the meta information, the meta fusion learner parameterizes a neural network as the fusion function for the input task. Namely, the meta fusion learner establishes a mapping from meta information to fusion parameters, while generalizing the learning-to-learn (meta-learning) mechanism to tasks. To parameterize a layer in the neural network, the meta fusion learner exploits a task-shared 3-D tensor to transform the meta in-

formation vector into the layer's weight matrix by multiplication. Mathematically, by regarding the 3-D tensor as a set of 2-D matrices, the transformation is actually a linear combination of these 2-D matrices weighted by the elements in meta information, thus satisfying the dynamic nature of the results. Moreover, considering the model complexity problem caused by excessive parameters, we adopt *Canonical Polyadic* (CP) Decomposition to lighten our model by decomposing the 3-D tensors into fewer parameters. In this way, MetaMMF can dynamically learn an item-specific fusion function for each item to integrate its multimodal features into the multimodal representation. After combining with the collaborative representation, the joint representation can then be directly used to predict the interactions between the users and micro-videos like MF or passed through an advanced model like GCN. We conduct extensive experiments on three publicly accessible datasets to verify the rationality and effectiveness of our MetaMMF method.

Overall, the three main contributions of our work are summarized as follows.

- We highlight the importance of dynamic multimodal fusion for effective representation learning in micro-video recommendation. Towards this end, we present a meta-learning-based method to fuse multimodal information of each item dynamically. This work represents one of the pioneering efforts in utilizing meta-learning for dynamic multimodal fusion.
- We devise a model MetaMMF for dynamic multimodal fusion. By analyzing the to-be-fused multimodal features of an item, MetaMMF dynamically parameterizes a neural network utilized as the item-specific fusion function. As a technical contribution, we innovatively employ tensor decomposition to downsize the model parameters, resulting in a lighter model with improved training efficiency.
- To validate the effectiveness of our proposed method, we conducted extensive experiments on three real-world datasets. The experimental results demonstrate that MetaMMF achieves state-of-the-art performance in enhancing multimodal representation learning.

5.2 Related Work

In this section, we review existing work on micro-video recommendation, multimodal fusion, and meta-learning, which are the most relevant to this work.

5.2.1 Micro-Video Recommendation

Consistent with the general personalized recommendation, the studies in this field parameterize users and micro-videos with embeddings, and reconstruct their interactions to learn parameters. Recently, multiple modalities of micro-videos are

widely exploited to enhance their representations in related literature because they can provide rich auxiliary information [101].

One of the early studies, Visual Bayesian Personalized Ranking (VBPR) [54] model, projects the modal features of each micro-video into a modal representation, then concatenated with the collaborative representation to form the final item representation. Such a fusion can only capture a linear relationship. With the rise of deep learning, many studies have successfully employed the non-linear fusion of multimodal features. For example, User-Video Co-Attention Network (UVCAN) [113] utilizes attention mechanisms at the user and micro-video levels to selectively incorporate modality information for representation learning. Nevertheless, its fusion method, a weighted sum of multimodal features, results in coarse-grained multimodal relationships. More recently, Graph Convolution Network (GCN) [49] has gained popularity for recommendation representation learning due to its outstanding performance. MMGCN [186] builds modality-aware GCNs based on a user-item bipartite graph to learn unimodal representations for users and micro-videos. However, its final multimodal fusion relies on a linear combination, neglecting the varying impact of different modalities. Heterogeneous Hierarchical Feature Aggregation Network (HHFAN) [14] leverages a modality-aware heterogeneous information graph to explore complex relationships among users, micro-videos, and multimodal information, generating high-quality representations. Nevertheless, the fusion of the neighbor's multimodal features in HHFAN is performed using a static aggregation network. Invariant Representation Learning (InvRL) [38] learns invariant representations that account for user attention and mitigate the influence of spurious correlations in micro-video recommendations. The learned invariant representations can consistently predict user-item interactions across different environments. Despite this innovation, its fusion module simply concatenates multimodal features.

Although these previous studies have shown performance improvements, their static fusion approach limits the capture of diverse multimodal relationships in micro-videos, resulting in suboptimal representations and negatively affecting recommendation performance. To address this limitation, we propose a novel dynamic multimodal fusion-based recommendation model, enhancing the representation learning of micro-video by dynamically fusing multimodal information. Our model utilizes the multimodal features of micro-videos and leverages meta-learning to dynamically parameterize a fusion neural network for each micro-video. This adaptive approach enables the capture of specific multimodal relationships pertaining to each micro-video. Furthermore, our model is designed to be model-agnostic and lightweight, providing distinct advantages in flexibility and efficiency compared to previous methods.

5.2.2 Multimodal Fusion

Multimodal fusion is one of the traditional topics in multimodal machine learning. In technical terms, multimodal fusion refers to integrating information from multiple

modalities to predict a class through classification or a continuous value through regression [6]. Most studies on multimodal fusion are model-agnostic, which can be divided into early (*i.e.*, feature-based), late (*i.e.*, decision-based), and hybrid fusion.

Early fusion integrates features extracted from different modalities. For example, ACNet [65] utilizes the attention mechanism to integrate valuable features from RGB and depth branches adaptively. However, the fusion process only considers a complementary relationship and utilizes a static layer with element-wise addition. In contrast, late fusion combines the output results of multiple models for multiple modalities. The related studies focus on fusing unimodal decision values, such as the neural multimodal cooperative learning (NMCL) [184] model, which enhances individual features in each modality using cooperative nets and performs late fusion on the prediction results from different modalities. However, these approaches overlook the feature-level interactions between modalities. Hybrid fusion, on the other hand, combines outputs from early fusion and individual unimodal predictors, aiming to leverage the advantages of both above-described methods in a common framework, such as Combine Early and Late Fusion Together (CELFT) [177]. This universal hybrid framework considers the fusion of different levels regardless of the fusion dynamics.

These fusion approaches adhere to an immutable formulation, *i.e.*, the fusion process remains constant regardless of the input. Such fusion strategies are unaware of the multimodal content of items. Considering that the relationships among modalities vary with the multimodal features of each item, we introduce a meta-learning-based multimodal fusion method. Our method dynamically generates item-specific fusion parameters by analyzing the input multimodal features, enabling us to achieve a superior fusion capability.

5.2.3 Meta-Learning in Recommendation

Meta-learning, also known as learning-to-learn, is inspired by the human learning process, which can quickly learn new tasks based on a small number of examples. Meta-learning tends to train a model that can adapt to a new task that is not used during the training with a few examples [86, 33]. The related studies can be classified into three types: metric-based [146], memory-based [137], and optimization-based meta-learning [40].

Metric-based methods learn a metric or distance function over tasks, while model-based methods aim to design an architecture or training process for rapid generalization across tasks. Lastly, optimization-based methods directly adjust the optimization algorithm to enable quick adaptation with just a few examples. Previous studies [146, 137] on metric-based and memory-based meta-learning convert the recommendation task to the classification problem. The work in [161] implements a meta-learning strategy by learning a neural network classifier whose biases are determined by the item history. The classifier performs recommendations by predicting whether a user consumes an item. A series of recent studies [221] have

adopted the optimization-based meta-learning approach and chose model-agnostic meta-learning (MAML) [40] for model training. For example, MeLU [86] applies the MAML framework to score the affinities, which rapidly adapts to new users or items based on sparse interaction history.

From the perspective of meta-learning theory, learning the multimodal fusion process of each micro-video can be regarded as an individual task with limited training samples. Inspired by this, we propose a meta-learning strategy to implement dynamic multimodal fusion for learning robust representation of each micro-video in the recommendation scenario.

5.3 Preliminaries

In this section, we first introduce the task of micro-video recommendation. We then shortly recapitulate the widely used multimodal representation learning based on neural networks, highlighting the limitation caused by using static multimodal fusion. Finally, we formalize the problem of dynamic multimodal fusion. For ease of reading, we use a bold uppercase letter to denote a matrix, a bold lowercase letter to denote a vector, an italic letter to denote a scalar, and a calligraphic uppercase letter to denote a set.

5.3.1 Micro-Video Recommendation

Micro-video recommendation focuses on inferring the preference degree of user u on micro-video item i. Basically, all the models can be composed of two parts: representation learning and interaction prediction [106]. The former is responsible for yielding vector representations $\mathbf{e}_u$ and $\mathbf{e}_i$ for user u and item i, respectively. Then the preference score of user u on item i can be estimated with the function,

$$\hat{y}_{ui} = \rho(\mathbf{e}_u, \mathbf{e}_i), \tag{5.1}$$

where $\rho(\cdot)$ denotes the prediction function (*e.g.*, inner product). Since each item is associated with multimodal (*e.g.*, visual, acoustic, and textual) features, many efforts have been made to exploit them to enhance the item representation. In recent studies, a typical operation is to represent item i by concatenating its collaborative representation and multimodal representation,

$$\mathbf{e}_i = \begin{bmatrix} \mathbf{e}_i^m \\ \mathbf{e}_i^c \end{bmatrix}, \tag{5.2}$$

where $\mathbf{e}_i^c \in \mathbb{R}^{d_c}$ indicates the collaborative representation bound up with ID, and $\mathbf{e}_i^m \in \mathbb{R}^{d_m}$ indicates the multimodal representation learned by integrating features

from multiple modalities. d_c and d_m denote the dimensions of the two kinds of representation, respectively.

5.3.2 Multimodal Fusion for Multimodal Representation Learning

In order to obtain multimodal representation, a procedure is required to integrate multimodal features into the same embedding space [99]. Multimodal fusion emerges to resolve how to combine the information from heterogeneous sources and represent information in a format that a computation model can work with. Therefore, it is widely applied to learning joint multimodal representations in micro-video recommendation. For item i, its multimodal representation $\mathbf{e}_i^m$ can be mathematically expressed as,

$$\mathbf{e}_i^m = f(\mathbf{x}_i^v, \mathbf{x}_i^a, \mathbf{x}_i^t), \tag{5.3}$$

where $\mathbf{x}_i^v \in \mathbb{R}^{d_v}$, $\mathbf{x}_i^a \in \mathbb{R}^{d_a}$, and $\mathbf{x}_i^t \in \mathbb{R}^{d_t}$ denote the features deriving from visual, acoustic, and textual modalities of item i, respectively. d_v, d_a, and d_t denote the dimensions of these modality features, respectively. $f(\cdot)$ denotes the multimodal fusion function. Considering the ability to learn complex decision boundaries, recent studies have widely adopted neural networks as the multimodal fusion function [125, 129]. For example, VBPR uses one neural layer to implement multimodal fusion by projecting the concatenation of the visual, acoustic, and textual modality features,

$$\mathbf{e}_i^m = \mathbf{W} \begin{bmatrix} \mathbf{x}_i^v \\ \mathbf{x}_i^a \\ \mathbf{x}_i^t \end{bmatrix}, \tag{5.4}$$

where $\mathbf{W} \in \mathbb{R}^{d_m \times (d_v + d_a + d_t)}$ denotes the weight matrix parameter. However, such multimodal fusion functions with static parameters cannot model the various relationships among multiple modalities in different micro-videos.

5.3.3 Problem Set-Up of Dynamic Multimodal Fusion

We focus on the dynamic multimodal fusion problem in micro-video recommendation. In this chapter, static fusion regards the multimodal fusion of all items as the same task. In contrast, we treat multimodal fusion for each item as an individual task in dynamic fusion. We aim to generate the item-specific parameters in a neural network format for each fusion task to achieve dynamic multimodal fusion. As discussed in Section 1, meta-learning is a potential solution due to its ability to learn parameters for multiple tasks and address the few-sample training problem for each task. In particular, we aim to develop a meta-learning model capable of processing an item's multimodal features as input, and generating neural network parameters

that can be utilized to fuse the different modalities of the item. Let $\mathcal{X}_i = \{\mathbf{x}_i^v, \mathbf{x}_i^a, \mathbf{x}_i^t\}$ be the set of multimodal features belonging to item i. The learning task of item i is to obtain its multimodal representation as the output of a neural network fusion function $f(\mathcal{X}_i|\Theta_i)$ where the parameters Θ_i are produced from the multimodal features to be fused:

$$\mathbf{e}_i^m = f(\mathcal{X}_i|\mathcal{G}(\mathcal{X}_i)). \tag{5.5}$$

The meta-learning refers to learning the function $\mathcal{G}(\mathcal{X}_i)$ that takes multimodal features $\mathcal{X}_i$ as input and produces the parameters of the neural network $f(\mathcal{X}_i|\Theta_i)$. In the following section, we describe the framework for learning $\mathcal{G}(\mathcal{X}_i)$ in detail.

5.4 Method

We first present the general MetaMMF framework as illustrated in Figure 5.2, elaborating on how to learn to parameterize a neural layer for fusing the multimodal features of a micro-video. Based on this, we then extend the single-layer structure into a multi-layer one. To lighten the space complexity of MetaMMF, we present a parameter simplification method based on tensor decomposition. Furthermore, we show that MetaMMF is model-agnostic and can be used as the early step prior to other representation learning functions, such as MF and GCN. Lastly, we provide the optimization of MetaMMF.

5.4.1 General One-Layer Fusion Framework

As discussed in Section 1, we view dynamic multimodal fusion from a meta-learning perspective, and regard the multimodal fusion of each micro-video as an individual learning task. Formulated as Eqn.(5.5), we propose a meta-learning model MetaMMF to learn a neural network for each task in terms of the micro-video's multimodal information, and thus yield an item-specific multimodal fusion function for representation learning. We first illustrate how MetaMMF dynamically learns the parameter of a neural layer, and then generalizes it to a multi-layer network.

Specifically, a meta-learning function $\mathcal{G}$ is responsible for generating the adaptive parameters of each fusion task in a neural network format by processing the meta information. Considering that multimodal features can reflect the relationships among modalities in micro-videos, we employ them as the source of meta information to decide their fusion parameters. In this way, the dynamic parameter generation of a one-layer fusion framework is defined as:

$$\mathbf{W}_i = \mathcal{G}(\mathcal{X}_i), \tag{5.6}$$

where $\mathbf{W}_i \in \mathbb{R}^{d_m \times (d_v + d_a + d_t)}$ is the weight matrix of a neural layer specific to item i, produced by the meta-learning function $\mathcal{G}$ from the multimodal features of item i. In

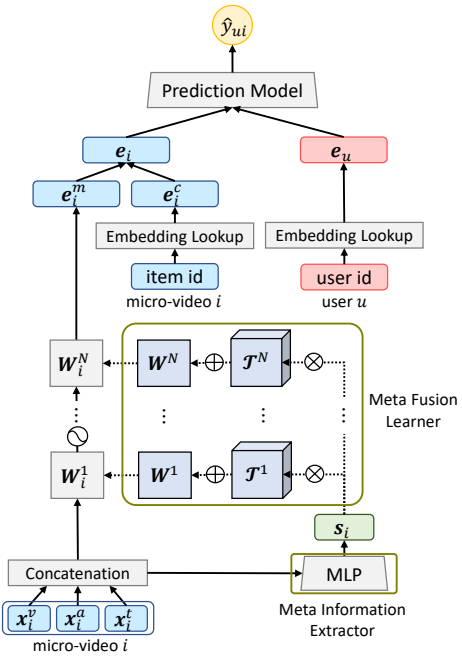


Fig. 5.2: Schematic illustration of our proposed model.

particular, the design of function $\mathcal{G}$ is twofold, including meta information extractor and meta fusion learner.

5.4.1.1 Meta Information Extractor

In practice, the multimodal features of micro-videos are highly dimensional, redundant, and noisy, which cannot be fed directly to the meta-learning function. Towards this end, meta information extractor is employed to extract informative and concise meta information from multimodal features of the target item. In this work, we adopt a multi-layer perceptron (MLP) for the extraction,

$$\begin{cases} \phi_{in}(\mathcal{X}_i) = \begin{bmatrix} \mathbf{x}_i^v \\ \mathbf{x}_i^a \\ \mathbf{x}_i^t \end{bmatrix} \\ \mathbf{s}_i = \phi_{out}\Big(\phi_{mid}\big(\phi_{in}(\mathcal{X}_i)\big)\Big) \end{cases}, \tag{5.7}$$

where ϕ_{in}, ϕ_{mid}, and ϕ_{out} respectively denote the input, hidden, and output layers of MLP, and each layer consists of its own weight matrix, bias vector, and activation

function. To be more specific, we uniformly select ReLU as the activation function. After successive multi-layer processing, $\mathbf{s}_i \in \mathbb{R}^{d_s}$ indicates the vector of meta information, wherein the size d_s is far less than the length of input feature $(d_v + d_a + d_t)$. The possibility of other extraction approaches that cannot be ruled out, such as the attention mechanism, will be further explored in future work.

5.4.1.2 Meta Fusion Learner

Meta fusion learner is responsible for rapidly parameterizing the multimodal function based on the task-dependent meta information. More specifically, meta fusion learner learns the mapping from the meta information $\mathbf{s}_i$, derived from the multimodal features $\mathcal{X}_i$, to the weight matrix parameter $\mathbf{W}_i$. Mathematically, the most direct and practicable way of mapping a 1-D vector into a 2-D weight matrix is to multiply by a 3-D tensor. Hence, in this work, we partially implement meta fusion learner as:

$$\mathbf{W}_i^* = \mathcal{T} \times_3 \mathbf{s}_i, \tag{5.8}$$

where $\mathcal{T} \in \mathbb{R}^{d_m \times (d_v + d_a + d_t) \times d_s}$ is the 3-D tensor parameter for meta fusion learner; d_m, $(d_v + d_a + d_t)$, and d_s indicate the argument values of height, width, and depth of $\mathcal{T}$. The operator $\times_3$ denotes that the multiplication is being performed along the tensor's third dimension (d_s). The tensor depth is exactly set equal to the size of meta information vector $\mathbf{s}_i$, in order to make the tensor-vector multiplication along the direction of depth. Thus, the product $\mathbf{W}_i^* \in \mathbb{R}^{d_m \times (d_v + d_a + d_t)}$ is dynamically adaptive to fusing the multimodal features of the target item i.

Beyond this, we also reserve an item-shared weight matrix $\mathbf{W} \in \mathbb{R}^{d_m \times (d_v + d_a + d_t)}$, which is equivalent to the previous static fusion method. This parameter contributes to modeling the elementary and common fusion for all the micro-videos. In comparison, the item-specific weight matrix $\mathbf{W}_i^*$ can be regarded as learning or adaption for the multimodal fusion task of item i. By combining $\mathbf{W}$ and $\mathbf{W}_i^*$, we obtain the parameter underlying a one-layer neural network for addressing the multimodal fusion task of a specific item. The multimodal representation can be learned via such a dynamically parameterized network as:

$$\begin{cases} \mathbf{W}_i = \mathbf{W} + \mathbf{W}_i^* \\ \mathbf{e}_i^m = \mathbf{W}_i \begin{bmatrix} \mathbf{x}_i^v \\ \mathbf{x}_i^a \\ \mathbf{x}_i^t \end{bmatrix} \end{cases}. \tag{5.9}$$

From the viewpoint of meta-learning, the weight $\mathbf{W}$ denotes the internal parameter that is broadly suitable to many tasks, while the generated parameter $\mathbf{W}_i^*$ can be regarded as fine-tuning the layer weight parameter for quickly adapting to an individual task [40].

5.4.2 Deep Multi-Layer Fusion Framework

Due to the multi-layer nature of deep neural networks, each successive layer is hypothesized to represent the information more abstractly. It is claimed that most functions that can be represented compactly by deep architectures cannot be represented by a compact shallow architecture [202]. Hence, proposing a multi-layer version of the MetaMMF model is necessary. In brief, multi-layer MetaMMF jointly learns the weight matrix parameters for a succession of neural layers, which are used to fuse the multimodal features step by step. Please kindly note that we omit the bias parameters in a neural layer, without which the performance is still good. Similarly, all the parameters are dynamically generated by taking in the multimodal features of the target item,

$$\{\mathbf{W}_i^n\}_{n=1}^N = \mathcal{G}(\mathcal{X}_i), \tag{5.10}$$

where N is the number of layers. Specifically, meta information $\mathbf{s}_i$ is likewise extracted from the concatenation of multimodal features as Eqn.(5.7). Subsequently, the weight matrices of all layers are determined as follows,

$$\{\mathbf{W}_i^n\}_{n=1}^N = \{\mathbf{W}^n + \mathcal{T}^n \times_3 \mathbf{s}_i\}_{n=1}^N, \tag{5.11}$$

where the 3-D tensors $\{\mathcal{T}^n\}_{n=1}^N$ and the 2-D matrices $\{\mathbf{W}^n\}_{n=1}^N$ are all trainable parameters for learning the weight matrices $\{\mathbf{W}_i^n\}_{n=1}^N$ of N layers, respectively. All the elements in the sequence of 3-D tensors $\{\mathcal{T}^n\}_{n=1}^N$ have consistent depths, which are equivalent to the length of the meta information vector $\mathbf{s}_i$. Obviously, multi-layer MetaMMF is an extension and stacking of the single-layer ones. In this way, each item can be provided with its individual multi-layer neural network for more complex multimodal fusion. The multimodal representation of item i can be learned via its own N-layer neural network as,

$$\mathbf{e}_i^m = \mathbf{W}_i^N \sigma\Bigg(\mathbf{W}_i^{N-1}\sigma\Big(...\sigma\big(\mathbf{W}_i^2\sigma(\mathbf{W}_i^1\begin{bmatrix}\mathbf{x}_i^v\\ \mathbf{x}_i^a\\ \mathbf{x}_i^t\end{bmatrix})\big)...\Big)\Bigg), \tag{5.12}$$

where σ denotes the activation function of the neural layer to implement the non-linearity of fusion, and we employ LeakyReLU in this work. In terms of the network structure design, we employ a tower pattern. The first hidden layer is the broadest, containing fewer neurons in each subsequent layer. As we progress through the layers, we reduce the number of neurons by half, ensuring that each layer has no less than the output layer size d_m. The rationale behind this design is that deeper layers with fewer hidden neurons can learn more abstract features from the data.

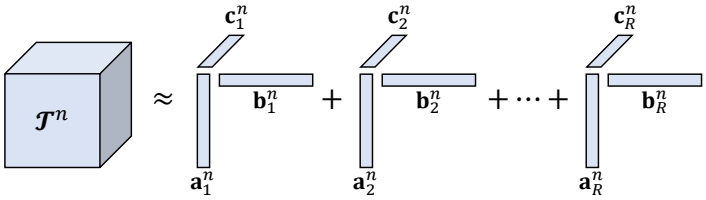


Fig. 5.3: CP decomposition of a 3-D tensor.

5.4.3 Model Simplification

Although the multi-layer framework can model a more complex fusion function, the number of parameters inevitably increases as the layers stack. Compared with a static neural network, multi-layer MetaMMF is equipped with extra 3-D tensor parameters for dynamic parameterization, increasing the storage cost and training time. To address this issue, we propose a simplification method using canonical polyadic (CP) decomposition to reduce the number of parameters in MetaMMF. Specifically, CP decomposition factorizes a tensor into a sum of component rank-one tensors, each of which is composed of the outer product of vectors. Through such decomposition, the dimension of the parameters can be greatly reduced. Therefore, we utilize CP decomposition to construct the 3-D tensor parameters used in MetaMMF. As illustrated in Figure 5.3, we can write the 3-D tensor $\mathcal{T}^n \in \mathbb{R}^{P\times Q\times Z}$ at the n-th layer as:

$$\mathcal{T}^n \approx \sum_{r=1}^{R} \mathbf{a}_r^n \circ \mathbf{b}_r^n \circ \mathbf{c}_r^n, \tag{5.13}$$

where notation $\circ$ denotes the outer product of vectors, R is a positive integer and $\mathbf{a}_r^n \in \mathbb{R}^P$, $\mathbf{b}_r^n \in \mathbb{R}^Q$, and $\mathbf{c}_r^n \in \mathbb{R}^Z$ for $r = 1, ..., R$. Elementwise, Eqn.(5.13) is written as,

$$t_{pqz}^n \approx \sum_{r=1}^{R} a_{pr}^n b_{qr}^n c_{zr}^n, p = 1, ..., P; q = 1, ..., Q; z = 1, ..., Z. \tag{5.14}$$

It is proven that the tensor $\mathcal{T}^n$ can be approximatively decomposed into three factor matrices. The factor matrices refer to the combination of the vectors from the rank-one components, *i.e.*, $\mathbf{A}^n = [\mathbf{a}_1^n\ \mathbf{a}_2^n\ \cdots\ \mathbf{a}_R^n]$ and likewise for $\mathbf{B}^n$ and $\mathbf{C}^n$. Using these definitions, we can replace the tensor $\mathcal{T}^n$ with the three matrices $\mathbf{A}^n \in \mathbb{R}^{P\times R}$, $\mathbf{B}^n \in \mathbb{R}^{Q\times R}$, and $\mathbf{C}^n \in \mathbb{R}^{Z\times R}$ to generate dynamic fusion parameters of the n-th layer as,

$$\mathbf{W}_i^n = \mathbf{W}^n + [\![\mathbf{A}^n, \mathbf{B}^n, \mathbf{C}^n]\!] \times_3 \mathbf{s}_i, \tag{5.15}$$

where $[\![\cdot, \cdot, \cdot]\!]$ indicates the computing process in Eqn.(5.13). Comparing the space complexity before and after CP decomposition, we can find it decreasing from $O(P \cdot Q \cdot Z)$ to $O(R \cdot (P + Q + Z))$, since $Q > P \gg R$. When the number of

tensors increases with fusion layers, the effect of model simplification will be more significant.

5.4.4 Model-Agnostic Combination with MF & GCN

As described above, MetaMMF can dynamically provide a multimodal fusion function in the form of a neural network for each item, which integrates the features from multiple modalities into a joint multimodal representation. Benefiting from the end-to-end training of neural networks, MetaMMF can be seamlessly combined with other representation learning frameworks where the joint multimodal representations are used for initialization. In this way, the combined model can take advantage of both multimodal and other useful information, such as collaborative filtering (CF) information and graph structure information. To demonstrate the model-agnostic characteristic of MetaMMF, we present two combined methods to address the micro-video recommendation task.

5.4.4.1 MetaMMF Plus MF

Matrix factorization (MF) is one of the most simple and effective recommendation methods, only utilizing CF signals. MF learns two independent embedding matrices to represent users and items, whose multiplication is used to reconstruct the observed ratings and predict the undiscovered ratings. The joint multimodal representation learned by MetaMMF can be directly injected into the MF framework, and thus combining the multimodal information with the CF signal. Therefore, we propose a combined method named **MetaMMF_MF**. Similar to VBPR [54], we extend the item collaborative representation $\mathbf{e}_i^c$ with the joint multimodal representation $\mathbf{e}_i^m$ as the complete item representation $\mathbf{e}_i$, like Eqn.(5.2). Following the MF framework, we estimate the affinity score between the target user and item by the inner product of their representations,

$$\hat{y}_{ui} = \mathbf{e}_u^\top \mathbf{e}_i = \mathbf{e}_u^\top \begin{bmatrix} \mathbf{e}_i^m \\ \mathbf{e}_i^c \end{bmatrix}, \tag{5.16}$$

wherein $\mathbf{e}_u \in \mathbb{R}^{(d_m+d_c)}$ and $\mathbf{e}_i^c \in \mathbb{R}^{d_c}$ are both trainable embedding parameters corresponding to user u and item i, responsible for capturing the CF signals.

5.4.4.2 MetaMMF Plus GCN

Graph Convolution Network (GCN) is a state-of-the-art representation learning technology that injects high-order connectivity signals into representations by recursively passing and aggregating messages based on the graph structure. In the recommendation task, GCN is always applied to the user-item interaction graph where the

nodes correspond to users and items, connected by historical interactions [102, 108]. Each node is initialized with an embedding, recursively updated by the convolution operation on its neighbor nodes. In most cases, the node initialization involves individual features, *e.g.*, identity and content features. Hence, nodes can finally obtain higher-quality representations by aggregating information from multi-hop neighbors.

However, such graph convolution operation only conducts on unimodal representation. For information of multiple modalities, some methods construct multiple interaction graphs for all modalities, such as MMGCN [186]. In comparison, MetaMMF can integrate the information from multiple modalities into a joint representation, directly serving as the node initialization of an interaction graph. Therefore, we employ MetaMMF in the initial layer of GCN, and propose a combined method named **MetaMMF_GCN**. Specifically, user u and item i can be initialized as:

$$\mathbf{e}_u^{(0)} = \mathbf{e}_u, \ \mathbf{e}_i^{(0)} = \begin{bmatrix} \mathbf{e}_i^m \\ \mathbf{e}_i^c \end{bmatrix} \tag{5.17}$$

where $\mathbf{e}_u^{(0)}$ and $\mathbf{e}_i^{(0)}$ denote the initial user and item representations at the 0-th layer of GCN. Similarly, $\mathbf{e}_u^{(0)}$ is an entirely trainable embedding of user u, while $\mathbf{e}_i^{(0)}$ is obtained by concatenating the joint multimodal representation derived from MetaMMF and the trainable collaborative representation of item i. Based upon the representations at the 0-th layer, we recursively formulate the item i's representation at the l-th layer as:

$$\mathbf{e}_i^{(l)} = \sigma(\mathbf{W}_1 \mathbf{e}_i^{(l-1)} + \mathbf{W}_2 \cdot \text{mean}_{u \in \mathcal{N}_i} \mathbf{e}_u^{(l-1)}), \tag{5.18}$$

where $\mathcal{N}_i$ denotes the neighborhood set of item i, *i.e.*, the users whom item i directly interacted with. σ denotes the LeakyReLU activation function. Note that we utilize the most common graph convolution operation in [49], and other more complicated choices are left to be explored. Analogously, we can obtain the representation $\mathbf{e}_u^{(l)}$ for user u by propagating information from its connected items. For convenience, we denote the final representations at depth L as $\mathbf{e}_u^{(L)}$ and $\mathbf{e}_i^{(L)}$. The inner product of the final user and item representations is used to predict their affinity score like most GCN-based methods do.

5.4.5 Optimization

In this work, we optimize the proposed models based on implicit feedback, such as clicks, views, and purchases. Compared to explicit ratings, implicit feedback is easier to collect but more challenging to model user preferences, due to its scarcity of negative feedback. To learn model parameters, we optimize the pairwise BPR loss [134], which is often used in recommender systems. It considers the relative order between observed and unobserved user-item interactions. Specifically, BPR assumes that the items in observed interactions reflect a user's preference more than

those in unobserved ones and should be assigned with higher prediction scores. The objective function is as follows,

$$\mathcal{L} = \sum_{(u,i,j)\in\mathcal{D}} -\ln \sigma(\hat{y}_{ui} - \hat{y}_{uj}) + \lambda||\Phi||_2^2, \tag{5.19}$$

where $\mathcal{D} = \{(u,i,j)|(u,i) \in \mathcal{R}^+, (u,j) \in \mathcal{R}^-\}$ denotes the pairwise training data, $\mathcal{R}^+$ indicates the observed interactions, and $\mathcal{R}^-$ is the unobserved interactions; $\sigma(\cdot)$ is the sigmoid function; Φ denotes all trainable model parameters, and λ controls the L_2 regularization strength to prevent overfitting. We adopt mini-batch Adam [77] to optimize the prediction model and update the model parameters. In particular, for a batch of randomly sampled triples $(u,i,j) \in \mathcal{D}$, we establish their representations after MetaMMF and the subsequent MF or GCN, and then update model parameters by using the gradients of the loss function.

5.5 Experimental Setup

In this section, the evaluation datasets are presented first, followed by an explanation of the experimental settings and an elaboration of baseline methods.

5.5.1 Datasets

As micro-videos contain rich multimodal information — frames, soundtracks, and descriptions, the experimental datasets should include such multimodal data for each item. Following MMGCN [186] and GRCN [185], we performed experiments on three publicly accessible datasets designed for micro-video personalized recommendation, including MovieLens[1], TikTok[2], and Kwai[3]. Table 5.1 summarizes the statistics of the datasets.

- *MovieLens:* This dataset is widely adopted to evaluate personalized recommendation. To adapt it for micro-video recommendation, we collected the descriptions of movies from the MovieLens-10M, and crawled the movies' trailers from Youtube[4]. Then, the pre-trained ResNet50 [53] extracted the visual features from these trailers' keyframes. FFmpeg[5] and VGGish [58] were adopted to separate audio tracks and learn the acoustic features, respectively. Sentence2Vector [3] was utilized to

[1] https://movielens.org/.
[2] https://www.tiktok.com/.
[3] https://www.kwai.com/.
[4] https://www.youtube.com/.
[5] http://ffmpeg.org/.

Table 5.1: Statistics of the evaluation datasets. (d_v, d_a, and d_t denote the dimensions of visual, acoustic, and textual modality feature data, respectively.)

Dataset	#Users	#Items	#Interactions	Density	d_v	d_a	d_t
MovieLens	55,485	5,986	1,239,508	0.37%	2048	128	100
TikTok	36,656	76,085	726,065	0.03%	128	128	128
Kwai	7,010	86,483	298,492	0.05%	2048	-	128

derive the textual features from movies' descriptions. Our experiments treated all ratings as the implicit feedback between the corresponding user-item pairs.

- *TikTok:* This dataset is released by the micro-video sharing platform Tiktok, which allows users to create and share micro-videos. It consists of users, micro-videos, and their interactions (*e.g.*, clicks). The micro-video features in each modality are extracted and published without providing the raw data. Particularly, the textual features are extracted from the micro-video captions.
- *Kwai:* As a micro-video service provider, Kwai released a sizable micro-video dataset. The dataset contains users, micro-videos, and user behavior records with timestamps. To evaluate the proposed method from implicit feedback, we collected some click records of the corresponding users and micro-videos in a certain period. The acoustic features are unavailable, in contrast to the datasets mentioned above.

For each dataset, we randomly split the historical interactions of each user into three folds: 80% for training, 10% for validation, and the rest 10% for testing. For each observed user-item interaction in the training set, we treated it as a positive instance. Then we adopted the negative sampling strategy to pair it with negative items the user did not interact with in the training set. This approach contrasts with that of [185], which used the entire dataset for negative sampling. To ensure a fair comparison, the negative sampling setup remains consistent across our methods and the compared baselines. The constructed triples are used for parameter optimization. The validation and testing sets are used to tune the hyper-parameters and evaluate the performance in the experiments, respectively.

5.5.2 Experimental Settings

Evaluation Metrics

For each user in the testing set, we regarded the items that the user did not interact with as the negative ones. Then each recommendation method predicts the user's preference scores over all items, except the ones used in the training set. To measure the performance of top-K recommendation and preference ranking, we ranked the items in descending order and adopted the widely-used evaluation metrics: Precision@K, Recall@K, HR@K (Hit Ratio) and NDCG@K (Normalized

Discounted Cumulative Gain). By default, we set $K = 10$ and 20, respectively. We reported the average metrics of all users in the testing set.

Model Implementation

We implemented the two combined models proposed above MetaMMF_MF and MetaMMF_GCN via the development tools Pytorch [6] and Pytorch Geometric [7]. The user/item representation size is fixed at 64. We optimized our model using the Adam optimizer with a batch size of 3,000, searching in batches of {500, 1,000, 2,000, 3,000}. Besides, we initialized all model parameters using the well-known Xavier approach. To tune the hyperparameters, we applied a grid search based on the results from the validation set. Specifically, we searched for the optimal learning rate from {1e-4, 5e-4, 1e-3, 5e-3} and ultimately set it to 1e-3. The coefficient λ of L_2 normalization was searched within {1e-8, 1e-7, $\cdots$, 1}, and the optimal value was determined to be 1e-6. As another critical model hyperparameter, the length of the meta information vector, denoted by d_s, was fine-tuned among the values of {2, 3, ..., 10}, and eventually fixed at 5. Moreover, if Precision@K on the validation set does not rise for ten successive epochs, early stopping is adopted. Experimentally, the optimal layer number for MetaMMF on the MovieLens dataset is 4. The sequence of neuron sizes from the input layer to the output layer is as follows: $2276 \rightarrow 1024 \rightarrow 512 \rightarrow 256 \rightarrow 32$. On the TikTok and Kwai datasets, the optimal layer number for MetaMMF is 1. The sequences of neuron sizes are $384 \rightarrow 32$ and $2176 \rightarrow 32$, respectively.

5.5.3 Baseline Comparison

To demonstrate the effectiveness of our method on micro-video recommendation, we compared it with the following state-of-the-art methods. These baselines can be briefly divided into three groups: MF-based (*i.e.*, VBPR), GCN-based (*i.e.*, GraphSAGE, NGCF, LightGCN, and GAT), and multimodal (MMGCN, GRCN, LATTICE, and InvRL) methods.

- **VBPR** [54]: This is a benchmark model in the multimodal recommendation. It incorporates content information with the MF framework to predict the interactions between users and items. For adapting to micro-video recommendation, we concatenated the multimodal features of micro-videos as content information.
- **GraphSAGE** [49]: With the trainable aggregation function, this model can pass the message along the graph structure and collect them to update each node's representation. It considers both the structure information and the distribution of node features in the neighborhood. For making a fair comparison, we integrated multimodal features as node features to learn the representations.

[6] https://pytorch.org/.

[7] https://pytorch-geometric.readthedocs.io/en/latest/.

- **NGCF** [173]: This model presents a novel recommendation framework to integrate the interaction information into the representation learning process. By exploiting the high-order connectivity from user-item interactions, the model encodes the CF signals into the representations. For a fair comparison, we fed the multimodal features of micro-videos into the framework to predict the user-item interactions.
- **LightGCN** [56]: This light yet effective model includes the essential ingredients of GCN for recommendation. This model adopts the simple weighted sum aggregator as a graph convolution operation and combines the representations obtained at each layer to form the final representation.
- **GAT** [162]: This GCN-based method is able to automatically learn and specify different weights to the neighbors of each node. With the learned weights, it alleviates the noisy information from the neighbors to improve the GCN performance.
- **MMGCN** [186]: This is a framework designed for micro-video recommendation. It allocates an independent GCN for each modality, which learns the modality-specific user preferences and item representations via the propagation of modality information on the user-item graph. The outputs of these GCNs are integrated as the final representations for recommendation.
- **GRCN** [185]: This is a state-of-the-art method for multimodal recommendation with implicit feedback. It adaptively adjusts the structure of the interaction graph according to the status of model training, instead of keeping the structure fixed. Based on the refined graph, it applies a graph convolution layer to distill informative signals on user preference.
- **LATTICE** [206]: This is an advanced method for multimodal recommendation, which discovers the item relationship about multimodal features to learn a graph structure. Along the graph structure, graph convolutions can aggregate informative high-order affinities from neighbors for each item. Finally, the model makes recommendation by combining downstream CF methods.
- **InvRL** [38]: This is a state-of-the-art method for invariant representation learning in multimodal recommendation. InvRL can eliminate spurious correlations using heterogeneous environments to learn consistent invariant item representations across different environments. These invariant representations are then used to make accurate predictions of user-item interactions.

Baseline Implementation. To ensure the consistency of baseline implementation, we adopted the publicly available implementations of GraphSAGE[8], NGCF[9], LightGCN[10], GAT[11], MMGCN[12], GRCN[13], LATTICE[14], and InvRL[15]. Without specification, the default size of user/item representation is 64. For all the GCN-based

[8] https://github.com/williamleif/GraphSAGE.

[9] https://github.com/xiangwang1223/neural_graph_collaborative_filtering.

[10] https://github.com/gusye1234/LightGCN-PyTorch.

[11] https://github.com/PetarV-/GAT.

[12] https://github.com/weiyinwei/MMGCN.

[13] https://github.com/weiyinwei/GRCN.

[14] https://github.com/CRIPAC-DIG/LATTICE.

[15] https://github.com/nickwzk/InvRL.

baselines, we uniformly set a two-layer structure with parameters initialized from Xavier, and used the Adam optimizer with a well-chosen mini-batch size for model optimization. The learning rate is tuned amongst {1e-4, 5e-4, 1e-3, 5e-3, 1e-2}.

5.6 Experimental Results

To validate the effectiveness of our proposed method, we conducted extensive quantitative and qualitative experiments to answer the following research questions:

- **RQ1**: Can our proposed method outperform the state-of-the-art baselines in the micro-video recommendation task?
- **RQ2**: How does the depth of fusion layers affect MetaMMF?
- **RQ3**: How do the representations benefit from the dynamic multimodal fusion?
- **RQ4**: How do the CP decomposition of parameters affect the convergence and the performance of MetaMMF?
- **RQ5**: How do the static weights affect MetaMMF?

5.6.1 Performance Comparison (RQ1)

Table 5.2: Overall performance comparison between our model and the baselines on MovieLens.

Methods	MovieLens							
	P@10	P@20	R@10	R@20	HR@10	HR@20	N@10	N@20
VBPR	0.0449	0.0348	0.1927	0.2875	0.1512	0.2340	0.1207	0.1494
MetaMMF_MF	0.0457	0.0353	0.1941	0.2912	0.1539	0.2375	0.1268	0.1562
GraphSAGE	0.0493	0.0363	0.1979	0.3049	0.1601	0.2445	0.1351	0.1653
NGCF	0.0539	0.0378	0.2132	0.3173	0.1658	0.2542	0.1366	0.1685
LightGCN	0.0557	0.0391	0.2318	0.3359	0.1753	0.2646	0.1525	0.1845
GAT	0.0564	0.0404	0.2250	0.3344	0.1813	0.2721	0.1523	0.1855
MMGCN	0.0569	0.0420	0.2310	0.3522	0.1900	0.2851	0.1535	0.1977
GRCN	0.0572	0.0424	0.2487	0.3558	0.1925	0.2867	0.1653	0.1999
LATTICE	0.0575	0.0426	0.2500	0.3562	0.1939	0.2871	0.1669	0.2010
InvRL	0.0575	0.0424	0.2518	0.3584	0.1934	0.2869	0.1666	0.2008
MetaMMF_GCN	**0.0599**	**0.0442**	**0.2613**	**0.3702**	**0.2018**	**0.2974**	**0.1757**	**0.2090**
% Improv.	4.17%	3.76%	3.77%	3.29%	4.07%	3.59%	5.27%	3.98%
p-value	1.12e-4	1.97e-4	1.85e-2	4.26e-3	2.38e-4	1.68e-3	3.05e-3	5.73e-3

Table 5.3: Overall performance comparison between our model and the baselines on TikTok.

Methods	TikTok							
	P@10	P@20	R@10	R@20	HR@10	HR@20	N@10	N@20
VBPR	0.0138	0.0111	0.0600	0.0945	0.0496	0.0799	0.0397	0.0500
MetaMMF_MF	0.0143	0.0119	0.0673	0.1062	0.0521	0.0854	0.0422	0.0531
GraphSAGE	0.0150	0.0124	0.0718	0.1134	0.0539	0.0893	0.0436	0.0560
NGCF	0.0172	0.0141	0.0845	0.1335	0.0617	0.1014	0.0513	0.0658
LightGCN	0.0184	0.0150	0.0921	0.1455	0.0705	0.1160	0.0565	0.0726
GAT	0.0187	0.0152	0.0933	0.1474	0.0707	0.1163	0.0573	0.0736
MMGCN	0.0186	0.0148	0.0927	0.1436	0.0706	0.1149	0.0570	0.0724
GRCN	0.0195	0.0162	0.0948	0.1480	0.0717	0.1162	0.0586	0.0745
LATTICE	0.0199	0.0164	0.0977	0.1535	0.0714	0.1177	0.0583	0.0754
InvRL	0.0203	0.0166	0.1002	0.1559	0.0726	0.1179	0.0612	0.0779
MetaMMF_GCN	**0.0217**	**0.0172**	**0.1065**	**0.1612**	**0.0775**	**0.1237**	**0.0652**	**0.0816**
% Improv.	6.90%	3.61%	6.29%	3.40%	6.75%	4.92%	6.54%	4.75%
p-value	3.71e-3	7.81e-3	1.65e-3	2.08e-2	6.64e-4	3.92e-3	3.17e-3	6.95e-3

Table 5.4: Overall performance comparison between our model and the baselines on Kwai.

Methods	Kwai							
	P@10	P@20	R@10	R@20	HR@10	HR@20	N@10	N@20
VBPR	0.0108	0.0088	0.0302	0.0441	0.0220	0.0359	0.0221	0.0283
MetaMMF_MF	0.0139	0.0102	0.0388	0.0536	0.0285	0.0418	0.0336	0.0384
GraphSAGE	0.0135	0.0105	0.0351	0.0490	0.0285	0.0429	0.0318	0.0364
NGCF	0.0138	0.0106	0.0378	0.0553	0.0282	0.0434	0.0334	0.0391
LightGCN	0.0153	0.0114	0.0417	0.0606	0.0314	0.0468	0.0357	0.0420
GAT	0.0146	0.0111	0.0419	0.0609	0.0299	0.0455	0.0360	0.0417
MMGCN	0.0140	0.0114	0.0431	0.0614	0.0309	0.0466	0.0361	0.0419
GRCN	0.0154	0.0116	0.0456	0.0649	0.0320	0.0479	0.0367	0.0428
LATTICE	0.0156	0.0117	0.0460	0.0656	0.0323	0.0482	0.0379	0.0435
InvRL	0.0159	0.0119	0.0463	0.0668	0.0324	0.0489	0.0381	0.0444
MetaMMF_GCN	**0.0166**	**0.0122**	**0.0480**	**0.0680**	**0.0340**	**0.0500**	**0.0398**	**0.0453**
% Improv.	4.40%	2.52%	3.67%	1.80%	4.94%	2.25%	4.46%	2.03%
p-value	3.44e-3	1.03e-2	4.50e-3	6.64e-3	1.05e-2	1.19e-2	4.51e-3	5.66e-3

Table 5.2, 5.3 and 5.4 presents the results of our methods and baselines over the experimental datasets. Besides, it reports the improvements and statistical significance test, which are calculated between our proposed method and the strongest baseline (highlighted with underline). From the comparison results, we noted the following findings:

- VBPR obtains poor performance on three datasets. This indicates that the static fusion function is insufficient to fit the distinct multimodal fusion of different micro-videos, and thus easily leads to suboptimal multimodal representations. Similarly under the MF framework, MetaMMF_MF consistently outperforms VBPR across all cases, demonstrating the rationality and effectiveness of dynamic multimodal fusion. Despite obtaining rather competitive performance, MetaMMF_MF is still slightly inferior to the GCN-based methods due to the limitation of MF.
- Compared with the MF-based methods, it is obvious that representation learning can be further boosted via aggregation and propagation on the graph structure, since GCN-based methods show better performance on micro-video recommendation. More specifically, NGCF generally outperforms GraphSAGE since it injects high-order connectivities among users and items into its propagation layers. LightGCN achieves certain improvements over NGCF after removing some useless nonlinear operations in the original model. Meanwhile, GAT obtains remarkable performance attributed to the attention mechanism which can specify the adaptive weights of neighbor nodes.
- As a model specifically designed for micro-video recommendation, MMGCN slightly outperforms the general GCN-based methods in most cases. One possible reason is that MMGCN sufficiently leverages the multimodal information by message passing of each modality and information interchange across modalities. Furthermore, GRCN adaptively adjusts the false-positive edges of the interaction graph to discharge the ability of GCN, thus outperforming MMGCN across all datasets. In addition to constructing the user-item graph for representation learning, LATTICE also creates modality-aware graphs that accurately capture the similarity of modal features between items, resulting in improved performance. Furthermore, the recently presented InvRL is confirmed to be the strongest baseline. InvRL is groundbreaking in that it learns invariant item representations to mitigate the influences of spurious correlations that are not adequately addressed by other methods.
- MetaMMF_GCN consistently yields the best performance across all datasets. In particular, the improvements of MetaMMF_GCN over the strongest competitor *w.r.t.* NDCG@10 are 5.27%, 6.54%, and 4.46% in MovieLens, TikTok, and Kwai, respectively. Additionally, we conducted one-sample t-tests, which reveal that the improvements of MetaMMF_GCN over the strongest baseline are statistically significant (p-value < 0.05). The primary difference between MetaMMF and baselines lies in whether the multimodal fusion is dynamic or static. Hence, we attribute the significant improvements to our dynamic fusion strategy. MetaMMF can effectively model the complex relationships among the modalities in different items during representation learning by dynamically learning the item-specific

fusion networks. This underscores the importance of dynamic multimodal fusion in micro-video recommendation.

- Moreover, MetaMMF_GCN significantly outperforms its MF version. It is reasonable since MetaMMF_GCN further boosts the multimodal representations learned from MetaMMF via the message passing function of GCN, while MetaMMF_MF directly uses the raw representations. This demonstrates that MetaMMF can be combined with other representation learning frameworks, and achieve better performance by exploiting the advantages of both frameworks to the full.

5.6.2 Effect of Layer Number on MetaMMF (RQ2)

Table 5.5: Effect of fusion neural layer numbers (Metrics: P, R).

Methods	MovieLens		TikTok		Kwai	
	P@10	R@10	P@10	R@10	P@10	R@10
Early_MF	0.0448	0.1923	0.0137	0.0639	0.0118	0.0326
Late_MF	0.0399	0.1730	0.0131	0.0598	0.0108	0.0304
Hybrid_MF	0.0407	0.1766	0.0134	0.0651	0.0117	0.0321
MetaMMF_MF-1	0.0443	0.1844	**0.0143**	**0.0673**	**0.0139**	**0.0388**
MetaMMF_MF-2	0.0447	0.1884	0.0134	0.0609	0.0122	0.0353
MetaMMF_MF-3	0.0455	0.1930	0.0128	0.0580	0.0119	0.0349
MetaMMF_MF-4	**0.0457**	**0.1941**	0.0124	0.0584	0.0110	0.0337
% Improv.	2.01%	0.94%	4.38%	3.38%	17.80%	19.02%
p-value	3.37e-4	1.02e-4	3.13e-4	2.38e-4	6.74e-3	3.54e-3
Early_GCN	0.0573	0.2446	0.0186	0.0925	0.0151	0.0406
Late_GCN	0.0568	0.2264	0.0188	0.0890	0.0140	0.0386
Hybrid_GCN	0.0572	0.2477	0.0189	0.0884	0.0151	0.0396
MetaMMF_GCN-1	0.0584	0.2554	**0.0217**	**0.1065**	**0.0166**	**0.0480**
MetaMMF_GCN-2	0.0592	0.2587	0.0214	0.1047	0.0160	0.0452
MetaMMF_GCN-3	0.0595	0.2590	0.0211	0.1043	0.0153	0.0439
MetaMMF_GCN-4	**0.0599**	**0.2613**	0.0211	0.1040	0.0151	0.0438
% Improv.	4.54%	5.49%	14.81%	15.14%	9.93%	18.23%
p-value	3.03e-4	1.97e-3	2.01e-3	1.66e-3	2.62e-3	3.99e-3

To verify the fusion advantage of MetaMMF, we compared it with neural network-based early fusion [125], late fusion [67], and hybrid fusion [84] methods. We connected the output of these fusion methods with MF and GCN, respectively. Additionally, we examined the impact of layer numbers on MetaMMF's performance, considering the crucial role played by the meta-learned neural layer in MetaMMF.

Table 5.6: Effect of fusion neural layer numbers (Metrics: HR, N).

Methods	MovieLens		TikTok		Kwai	
	HR@10	N@10	HR@10	N@10	HR@10	N@10
Early_MF	0.1508	0.1204	0.0495	0.0392	0.0242	0.0282
Late_MF	0.1341	0.1120	0.0471	0.0391	0.0219	0.0225
Hybrid_MF	0.1370	0.1141	0.0482	0.0401	0.0239	0.0286
MetaMMF_MF-1	0.1491	0.1208	**0.0521**	**0.0422**	**0.0285**	**0.0336**
MetaMMF_MF-2	0.1516	0.1228	0.0515	0.0414	0.0249	0.0318
MetaMMF_MF-3	0.1534	0.1252	0.0482	0.0379	0.0243	0.0283
MetaMMF_MF-4	**0.1539**	**0.1268**	0.0462	0.0307	0.0235	0.0272
% Improv.	2.06%	5.32%	5.25%	5.24%	17.77%	17.48%
p-value	4.90e-4	6.09e-4	6.76e-4	5.47e-4	1.23e-2	7.05e-3
Early_GCN	0.1929	0.1654	0.0704	0.0568	0.0310	0.0346
Late_GCN	0.1837	0.1595	0.0675	0.0539	0.0287	0.0324
Hybrid_GCN	0.1925	0.1671	0.0678	0.0543	0.0309	0.0348
MetaMMF_GCN-1	0.1965	0.1709	**0.0775**	**0.0652**	**0.0340**	**0.0398**
MetaMMF_GCN-2	0.1992	0.1731	0.0767	0.0644	0.0327	0.0380
MetaMMF_GCN-3	0.2003	0.1736	0.0756	0.0640	0.0314	0.0375
MetaMMF_GCN-4	**0.2018**	**0.1757**	0.0757	0.0630	0.0308	0.0374
% Improv.	4.61%	5.15%	10.09%	14.79%	9.68%	14.37%
p-value	7.52e-4	2.59e-3	4.58e-3	5.64e-3	1.87e-3	5.43e-3

In particular, we varied the model depth in the $\{1, 2, 3, 4\}$ range. The experimental results of these methods are summarized in Table 5.5 and 5.6, where MetaMMF_MF-3 and MetaMMF_GCN-3 denote our two methods with three meta-learned fusion layers, and similar notations are used for the others. By analyzing the results presented in Table 5.5 and 5.6, we have made the following findings:

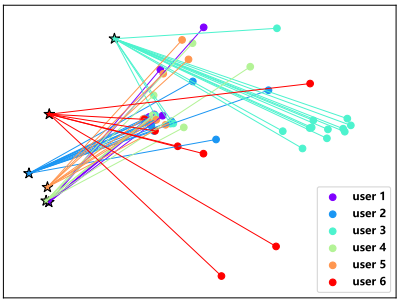
- On the datasets with less sparsity (*e.g.*, MovieLens), increasing the depth of MetaMMF substantially improves the performance when combined with both MF and GCN. The phenomenon validates that multi-layer MetaMMF is capable of modeling fine-grained multimodal fusion for items and thus enhances representation learning. On the sparse datasets (*e.g.*, TikTok and Kwai), increasing the number of meta-learned fusion layers has no positive effect on the performance of MetaMMF. Especially when combined with MF, one-layer fusion achieves the best performance, which is consistent with the prior VBPR [54]. The probable reason is that MF easily suffers from over-fitting with sparse data, and redundant fusion layers actually make this problem worse. When followed by GCN, stacking fusion layers also has little impact on model performance. This is because the message passing of GCN can effectively alleviate the data sparsity problem with just one fusion layer [173].

- Among the three kinds of fusion methods, early fusion consistently outperforms late fusion. Early fusion is the feature-level fusion that directly integrates multimodal features to learn the representation. By contrast, late fusion is at the decision level, overlooking the feature-level fusion among the modalities. This shows that late fusion is inapplicable to micro-video recommendation based on representation learning. Although the hybrid fusion method exploits both fusion methods in a common framework, it is inferior to the single early fusion in some cases. The possible reason is that the late fusion part drags down the performance.
- We performed one-sample t-tests on various datasets and metrics to compare the best fusion strategies (highlighted with underlines) with our methods that employ optimal layers (highlighted in bold). The results are recorded in Table **??**, where a p-value ¡ 0.05 indicates a statistically significant improvement. In the MF framework, we observed that MetaMMF significantly outperforms the early, late, and hybrid fusion methods when utilizing the optimal number of fusion layers. On the other hand, in the GCN framework, MetaMMF consistently outperforms all other fusion strategies, regardless of the number of fusion layers employed. This suggests that MetaMMF is better suited for collaborating with complex models, such as GCN, than other static fusion strategies. Our findings confirm the effectiveness of MetaMMF, demonstrating that dynamic multimodal fusion can significantly enhance the micro-video recommendation task.

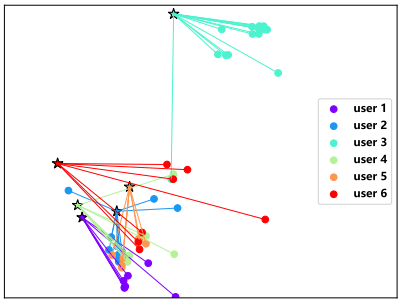
5.6.3 Effect of Dynamic Multimodal Fusion (RQ3)

In this section, we attempt to understand how our dynamic multimodal fusion strategy facilitates representation learning in the embedding space. Therefore, we randomly selected six users from TikTok associated with their relevant items. Note that the items are from the test set, which are not paired with users in the training phase. By employing the t-Distributed Stochastic Neighbor Embedding (t-SNE) [120] in 2-dimension, Figures 5.4(a)-(d) visualize the representations derived from VBPR, MetaMMF_MF, InvRL, and MetaMMF_GCN, respectively. By analyzing how their representations distribute, we have two key observations:

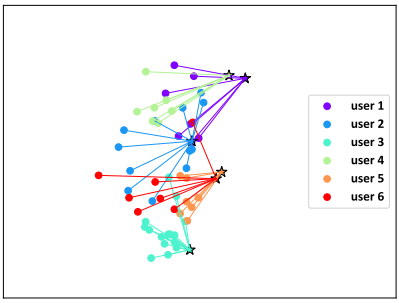
- The connectivities of users and items are well reflected in the embedding space, *i.e.*, embedded into the nearest parts of the space, resulting in discernible clustering. Compared to VBPR and InvRL, our MetaMMF_MF and MetaMMF_GCN models exhibit even more noticeable clustering, with points of the same color (*i.e.*, items interacted with by the same users) forming clusters. To evaluate the clustering performance of our methods and baselines, we computed their mean silhouette coefficients (S) for all samples shown in Figure 5.4. This coefficient measures the effectiveness and reasonableness of the clustering, with a value ranging between -1 and 1. A larger value indicates that samples from the same class are closer, while samples from different classes are farther apart, implying better clustering performance. Our methods' representations achieve better clustering performance



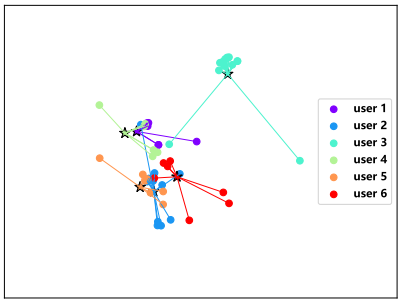
(a) VBPR ($S = -0.0624$)



(b) MetaMMF_MF ($S = 0.0272$)



(c) InvRL ($S = 0.0753$)



(d) MetaMMF_GCN ($S = 0.1248$)

Fig. 5.4: The visualization depicts the t-SNE transformed representations obtained from our methods and baselines. Each star corresponds to a user from the TikTok dataset, while points with the same color signify relevant items. A link between a star and a point represents their interaction. For optimal viewing, please refer to the colored version. The notation S indicates the mean silhouette coefficient of the clustering result for the sample representations.

than the competitors, as seen in the S values in Figure 5.4. This indicates that our MetaMMF model enhances representation learning effectively. We attribute this success to our model's dynamic multimodal fusion, which utilizes and integrates items' multimodal features in an adaptive approach.

- Jointly analyzing the same users across Figure 5.4, we find that the embeddings of their historical items tend to be closer in our methods. Meanwhile, the dynamic multimodal fusion prompts the item embeddings to distribute more discriminately in the visualization. It verifies that our proposed methods have an advantage in capturing the user-item correlation.

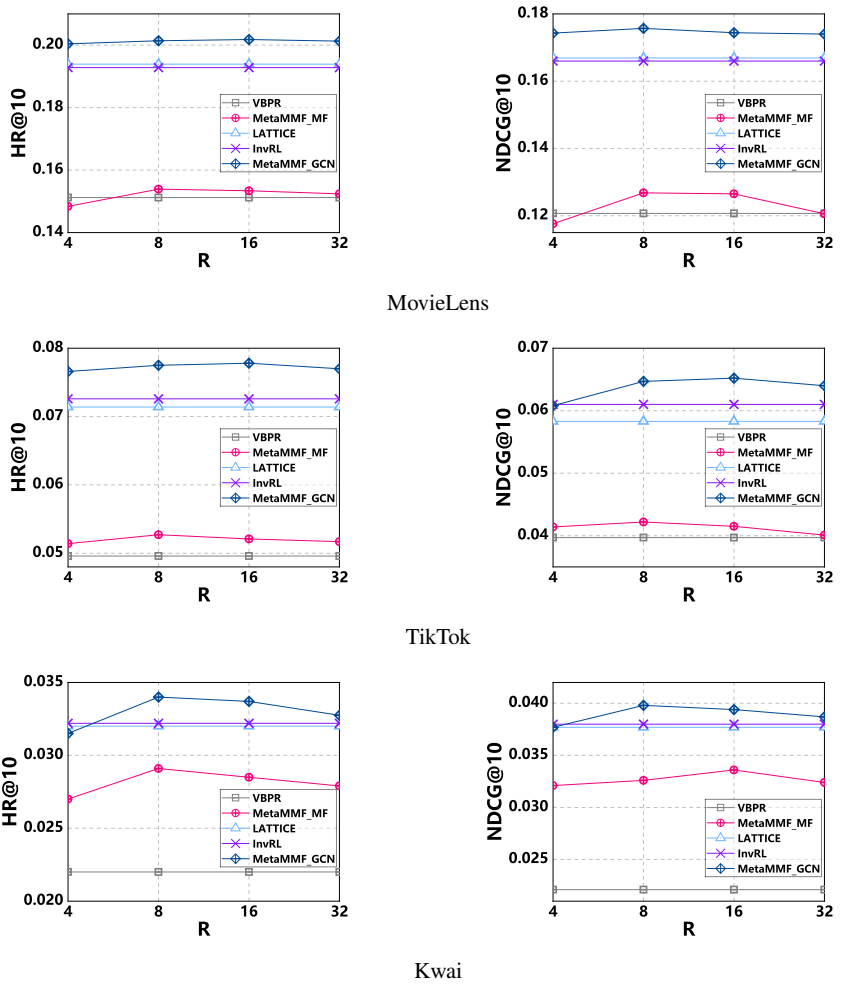


Fig. 5.5: Effect of hyper-parameter R in CP decomposition.

5.6.4 Effect of Parameter Decomposition (RQ4)

We investigate how the parameter decomposition (*i.e.*, CP decomposition of 3-D tensor parameters) affects model MetaMMF from two aspects of performance and convergence. In particular, we first study how the hyper-parameter R (*i.e.*, the number of components in Eqn.(5.13)) from CP decomposition affects the performance, and then explore the influence of CP decomposition on training epochs.

Effect of hyper-parameter R

Figure 5.5 shows the performance of our methods MetaMMF_MF and MetaMMF_GCN

with different numbers of decomposed components, as well as their most competitive baselines, against HR@10 and NDCG@10 evaluation protocols on different datasets. The value of hyper-parameter R is varied in the range of $\{4, 8, 16, 32\}$, while the number of fusion layers is optimal. From Figure 5.5, we can observe that:

- The hyper-parameter R in CP decomposition indeed affects the performance of our methods. Generally, a small decomposition value leads to the worst performance across all the datasets, since such large-scale decomposition cannot restore the fitting ability of the original 3-D tensor parameters. By increasing the value of R, the CP decomposition can offer better performance while effectively reducing our model's space complexity. However, blindly increasing the R value will not further enhance the performance, but instead burdening the model. The possible reason is that R has already been higher than the rank of the original 3-D tensor, and CP decomposition provides excess capacity.
- Despite the influence of R, our methods are consistently superior to VBPR and LATTICE, respectively. This demonstrates that CP decomposition can replace the original 3-D tensor parameters, and retain the ability of dynamic multimodal fusion within our model. With regard to the optimal value of R, we find that MetaMMF_MF and MetaMMF_GCN always achieve the best performance on datasets when the value of R is set to 8 or 16. In such a case, CP decomposition is able to reduce the model to nearly one percent of its original size. Hence, although MetaMMF uses 3-D tensor parameters to replace the 2-D matrix parameters in neural layers, the number of model parameters does not exceed by a wide margin attributing to CP decomposition.

5.7 Conclusion

In this work, we upgraded the multimodal fusion from static to dynamic to enhance the representation learning in the micro-video recommendation. We devised a novel meta-learning model MetaMMF, which realizes the target by learning the item-specific multimodal fusion functions for different items. The core of MetaMMF is the meta fusion learner which can adaptively parameterize a neural network for an item's multimodal fusion based on the meta information derived from its multimodal features. Extensive experiments on three real-world datasets demonstrate the rationality and effectiveness of leveraging dynamic fusion to learn multimodal representations. In the future, we will exploit MetaMMF to dynamically integrate user preferences on multiple modalities and produce higher-quality user representations, which will be beneficial to recommendation performance. Moreover, we are interested in introducing MetaMMF to address the cold-start problem of multimodal recommendation [37].

This work is an initial attempt at dynamic multimodal fusion, limited to a neural network framework and the micro-video recommendation scenario. Specifically, there are many other types of fusion functions, such as multiple kernel learning meth-

ods [48] and graphical models. Furthermore, multimodal fusion is widely applied in various tasks, such as emotion recognition [32], multimedia event detection [84], and multimedia retrieval. We expect the potential of dynamic multimodal fusion can be further explored towards other model structures and tasks.

Chapter 6
Attribute-driven Disentangled Representation Learning for Multimodal Recommendation

6.1 Introduction

In addition to aiding recommender systems in effectively capturing user preferences and improving recommendation accuracy, as introduced in Chapter 3 and Chapter 4, multimodal data also holds significant potential for enhancing the interpretability and controllability of recommendation results, which is crucial for ensuring user satisfaction. In this chapter, we investigate how disentangled representation learning of multimodal information can be utilized to achieve interpretable and controllable outputs in recommender systems.

It is well recognized that the entangled representations of users and items, which are utilized by the majority of existing recommendation methods, fail to directly capture fine-grained user preferences across diverse factors [118, 174]. This limitation constrains both the effectiveness and interpretability of recommender systems. In recent years, the study of disentangled representation learning has garnered significant attention in diverse fields, notably in computer vision [59, 115, 64], due to its capability to identify and disentangle the underlying factors behind data. Empirical evidence suggests that disentangled representations exhibit enhanced robustness, particularly in complex application contexts. Hence, many recommendation methods adopt disentangled representation learning techniques to learn robust and independent representations, ultimately enhancing recommendation performance [118, 126, 174, 159, 99]. For example, Ma et al. [118] employed disentangled representations to capture user preferences regarding different concepts associated with user intentions. Wang et al. [174] introduced a GCN-based model that produces disentangled representations by modelling a distribution over intents for each user-item interaction, exploring the diversity of user intentions on adopting items. However, these methods only concentrate on disentangling the user and item representations based on their ID embeddings. To exploit the difference between the factors behind data of various modalities, Liu et al. [99] estimated the users' attention weight to underlying factors of different modalities, utilizing a sophisticated attention-driven module.

F. Liu et al., *Multimodal Learning toward Recommendation*,
https://doi.org/10.1007/978-3-031-83188-1_6

Despite the considerable advancements brought about by disentangling techniques in recommender systems, existing studies often disentangle both users and items into latent factors. This methodology inherently constrains the model's interpretability and controllability. For example, consider using disentangled representations to elucidate the diverse factors, such as *style*, *brand*, *popularity*, and *price*, that shape user preferences in dress selection. The inherent abstraction of latent factors from current disentangling methods makes it difficult to pinpoint which factor represents each specific dress attribute. In particular, one factor might be loosely related to a combination of *brand* and *price*, while another might represent a mixture of *style* and *popularity*. This ambiguity in the latent factors limits the interpretability of the recommendation system, making it difficult to understand why a particular item was recommended to a user. Moreover, this lack of clarity also affects the controllability of the recommendation system. Suppose a user wants to receive recommendations specifically for dresses without considering her preferences for *price* and *popularity*. It would be challenging to adjust the recommender system to focus on those specific preferences, as the latent factors are not clearly tied to these attributes.

Item attributes manifested in various modalities can enrich the recommender system by offering diverse and complementary information. For example, the textual data might explicitly mention the *brand* or *price*, while visual data can reveal visual attributes, such as the *category* of items. Additionally, the *popularity* of items can be derived from the statistical information of interaction data. In fact, attributes represent specific, meaningful properties or characteristics of items. Consequently, using attributes to guide the disentanglement process is a promising way to improve the interpretability and controllability of conventional multimodal recommendation methods. In this chapter, we propose an **A**ttribute-**D**riven **D**isentangled **R**epresentation **L**earning method (AD-DRL for short), which disentangles factors in user and item representations across various modalities at different levels of attribute granularity. To obtain robust and independent representations for each factor associated with a specific attribute, we first disentangle the representations of features within and across different modalities. This process is guided by high-level attributes (e.g., *category* and *popularity*), which help reveal the underlying relationships between factors. Following this, we further enhance the robustness of the representations by fusing the multimodal features of the same factor. This step involves exploiting the relationships among representations of the same factor by leveraging low-level attributes (e.g., *category* attribute values for a clothing dataset, such as *jeans*, *jackets*, and *dresses*), resulting in finer-grained and more comprehensive disentangled representations. To validate the effectiveness of our method, we conduct extensive experiments and ablation studies on three real-world datasets. Experimental results demonstrate the superiority of our method AD-DRL compared to existing methods and showcase its capability in terms of interpretability and controllability.

In summary, the contributions of this chapter are threefold:

- In this chapter, we highlight the limitations of traditional disentangled representation learning in multimodal recommendation systems. To overcome these shortcomings, we introduce AD-DRL, which improves interpretability and con-

trollability by employing attributes to disentangle factors in user and item representations.
- To achieve robust and independent representations, we assign a specific attribute to each factor in multimodal features and disentangle factors at both the attribute and attribute-value levels.
- We conduct extensive experiments on three real-world datasets to validate the effectiveness of our method. The experimental analysis demonstrates the interpretability and controllability of our model.

The rest of this chapter is organized as follows: Section 6.2 provides a comprehensive review of related work. Section 6.3 introduces the preliminaries relevant to our proposed model. Section 6.4 details the components of the proposed model. Experimental results are presented in Section 6.5. Finally, Section 6.6 concludes the chapter.

6.2 Related Work

6.2.1 Interpretability of Recommender Systems

Explainable recommendation refers to personalized recommendation algorithms that not only generate recommendations but also provide explanations for why certain items are suggested. Early explainable recommendation systems primarily used collaborative filtering to identify patterns and similarities between users and items based on past interactions [144, 167]. For instance, the Explicit Factor Model aligns latent dimensions with explicit features to facilitate explainable recommendations. With the advancement of knowledge graphs (KGs) and graph neural networks, KG-based recommendation systems exhibit significant advantages in terms of interpretability due to the rich information about users and items encapsulated within the graph structures [18, 94, 68]. For example, Ma et al. [119] proposed a joint learning framework that integrates explainable rule induction in knowledge graphs with a rule-guided neural recommendation model. Numerous studies have also incorporated attention mechanisms into recommendation tasks to enhance recommendation performance [141, 188]. DAttn [141], for instance, introduces two word-level attention mechanisms—namely, local attention and global attention—based on a convolutional neural network to achieve more precise semantic representations and improve the interpretability of the model. With the success of transformers and Large Language Models (LLMs) in the natural language domain, several methods have begun leveraging these technologies to generate more direct and coherent explanatory text for recommendation [91, 92]. To harness the sentiment analysis capabilities of LLMs, SAGCN [104] proposes a chain-based prompting strategy to uncover semantic aspect-aware interactions which provide clearer insights into user behaviors at a fine-grained semantic level.

In contrast to the methods discussed above, the approach proposed in this chapter provides explicit explanations for recommendation results from the perspective of item attributes, clearly articulating the influence of various attributes in the recommendation process.

6.2.2 Controllability of Recommender Systems

The controllability of a recommender system refers to the ability of users to participate in the recommendation process by providing various forms of input, which has been shown to enhance their satisfaction with and trust in the recommendation results [79, 72, 22, 153, 212, 178]. Existing controllable methods generally falls into two categories: user controls during the preference estimation and the controls over recommendation lists [138, 11, 139]. In the first category of controllable methods, users can control the output of the recommender system in various ways, such as through interactive conversation [131, 149] or preference forms [60]. For example, Knijnenburg et al. [80] proposed an interaction method for an attribute-based recommender system where users specify the weights assigned to different item attributes to express their preferences. Hijikata et al. [60] implemented a music recommender system that allows users to intervene in the recommendation process by selecting preferred item categories or editing their profiles. Nie et al. [131] further suggested that users can modify the output of recommender systems to better align with their needs through dialogue-based interactions. The second category of controllable methods enables users to re-rank recommendation lists according to their preferences after the recommender system has returned the initial results to align with their expectations [51]. For instance, Schafer et al. [138] proposed a meta-recommender system that allows users to personalize the generation of a recommendation list by combining data from multiple sources and using various recommendation techniques. Hijikata et al. [150] demonstrated that user intervention itself increases user satisfaction, with higher levels of intervention leading to greater user satisfaction with the recommendations.

While the aforementioned methods represent initial efforts in enhancing the controllability of recommender systems, they primarily adopt on a user study perspective rather than a machine learning perspectives. Furthermore, they lack explicit explanations linking user actions to recommendation outcomes, thereby undermining trust in controllable recommender systems. In contrast to these existing approaches, our work seeks to disentangle user and item representations based on attributes, allowing users to control recommendation outcomes in accordance with their preferences for different attributes, thereby enhancing the interpretability of their actions.

6.3 Preliminaries

6.3.1 Problem Setting

Given a set of users $\mathcal{U} \in \{u\}$ and items $\mathcal{I} \in \{i\}$, we utilize three types of information to learn user and item representations: (1) **A user-item interaction matrix $\boldsymbol{R}$**, where each entry $r_{ui} \in \boldsymbol{R}$ represents the implicit feedback (*e.g.*, clicks, likes or purchases) of user u to item i. $r_{ui} = 1$ indicates that user u has interacted with item i and $r_{ui} = 0$ indicates that there is no interaction between user u and item i in the observed data; (2) **Multimodal features**, which mainly consist of two types of features associated with items, namely *reviews* and *images*; and (3) **Attribute information**, consisting of K attributes and their associated attribute values of items, used as supervision for disentangled representation learning. We aim to learn robust representations of users and items, thereby predicting a user's preference for items they have not yet interacted with.

6.3.2 Notations

Similar to previous work [185, 99], each user u and item i are assigned with a unique ID and respectively represented by a vector $\boldsymbol{v}_u \in \mathbb{R}^d$ and $\boldsymbol{v}_i \in \mathbb{R}^d$, which are randomly initialized in our model. For the review and image information, we use the BERT [35] and the ViT [81] to extract the raw textual features $\boldsymbol{e}_t \in \mathbb{R}^{d_0}$ and visual features $\boldsymbol{e}_v \in \mathbb{R}^{d_0}$, respectively. To tailor the recommendation-oriented features, we employ two non-linear transformations to cast $\boldsymbol{e}_t$ and $\boldsymbol{e}_v$ into the same feature space:

$$\begin{aligned} \boldsymbol{v}_t &= \sigma(\boldsymbol{W}_t \boldsymbol{e}_t + \boldsymbol{b}_t), \\ \boldsymbol{v}_v &= \sigma(\boldsymbol{W}_v \boldsymbol{e}_v + \boldsymbol{b}_v), \end{aligned} \tag{6.1}$$

where $\boldsymbol{W}_t, \boldsymbol{W}_v \in \mathbb{R}^{d \times d_0}$ and $\boldsymbol{b}_t, \boldsymbol{b}_v \in \mathbb{R}^d$ denote the weight matrices and bias vectors for textual and visual modality, respectively. $\sigma(\cdot)$ is the activation function.

Following the disentangled representation learning process in previous work [174, 99], we split the feature vector into K chunks, each corresponding to a specific item attribute (*e.g.*, *price*, *brand*). For simplicity, we equally split each modality's representation into K continuous chunks. Take item ID embedding as an example:

$$\boldsymbol{v}_i = (\boldsymbol{v}_i^1, \boldsymbol{v}_i^2, \cdots, \boldsymbol{v}_i^K), \tag{6.2}$$

where $\boldsymbol{v}_i^k \in \mathbb{R}^{\frac{d}{K}}$ is the item ID embedding corresponding to the k-th attribute. Analogously, $\boldsymbol{v}_t = (\boldsymbol{v}_t^1, \boldsymbol{v}_t^2, \cdots, \boldsymbol{v}_t^K)$, $\boldsymbol{v}_v = (\boldsymbol{v}_v^1, \boldsymbol{v}_v^2, \cdots, \boldsymbol{v}_v^K)$ and $\boldsymbol{v}_u = (\boldsymbol{v}_u^1, \boldsymbol{v}_u^2, \cdots, \boldsymbol{v}_u^K)$ are defined as the embeddings of textual feature, visual feature and user ID embedding, respectively.

6.3.3 Intuition of Attribute-driven Disentanglement

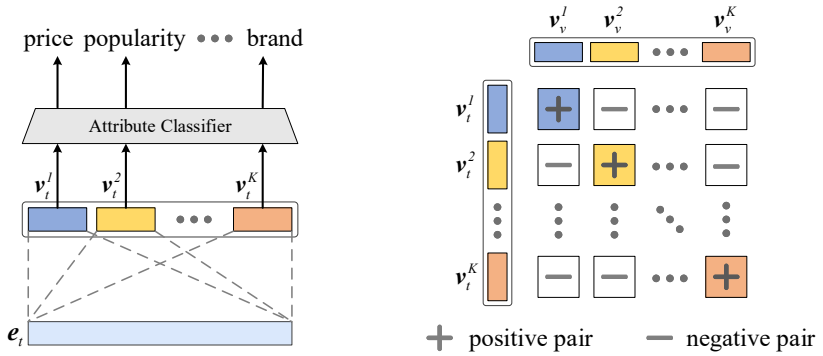
Our work not only aims to recommend accurate items to users but also advocates for attribute-driven disentanglement in multimodal recommender systems to enhance the interpretability and controllability of recommendations. Specifically, unlike the existing method that disentangles the latent factors in an unsupervised manner, we leverage the semantic labels of item attributes to learn attribute-specific sub-spaces for each attribute to obtain disentangled representations. In this work, the vector of each modality is composed of K chunks, with the assumption that each chunk is associated with an attribute (such as *price* or *brand*). The achievement of attribute-driven disentangling requires two necessary conditions. Firstly, each chunk should have a clear attribute reference, and chunks associated with different attributes should be distinguishable. For example, chunk j is associated with the *price*, while chunk k represents the vector of the *brand*. Secondly, each chunk of a specific attribute should accurately capture the specific attribute value. For example, chunk j should reflect the price level (i.e., expensive or cheap) of the product.

6.4 Attribute-driven Disentangled Representation Learning

In this section, we elaborate on our proposed model, termed AD-DRL, an acronym for *Attribute-Driven Disentangled Representation Learning model*. We aim to improve the interpretability and controllability of recommendation models by assigning a specific attribute to each factor in multimodal features. Specifically, AD-DRL comprises two disentangling modules: high-level and low-level attribute-driven disentangled representation learning. In the *high-level attribute-driven disentangled representation learning* module, we exploit the difference between attribute factors within the same modality feature and the consistency of the same attribute factor across different modalities. In contrast, the *low-level attribute-driven disentangled representation learning* module leverages the intrinsic relationships between items sharing the same attribute value.

6.4.1 High-Level Attribute-driven Disentangled Representation Learning

To obtain robust and independent representations for each attribute factor in multimodal features, AD-DRL enables disentangling within each modality feature and ensures consistency of representations across modalities. Next, we detail the intra-modality disentanglement and inter-modality disentanglement.



(a) Intra-Modality Disentanglement (b) Inter-Modality Disentanglement

Fig. 6.1: High-level attribute-driven disentangled representation learning module of our proposed AD-DRL. To disentangle attribute factors in multimodal features, (a) Intra-modality disentanglement module exploits the difference between attribute factors (e.g., *price*, *brand*, *category* and *popularity*) within the same modality feature, and (b) Inter-modality disentanglement module utilizes the consistency of the same attribute factor in different modality features.

Intra-Modality Disentanglement

After obtaining the feature vectors of each modality, we split these vectors into several chunks. Nevertheless, different attribute factors are still entangled within the chunks. In other words, a chunk may contain both *price*- and *brand*-related features. Therefore, in order to disentangle attribute factors in each modality feature, we employ an attribute classifier to nudge each chunk to predict the corresponding attribute, as shown in Figure 6.1a. Taking the k-th chunk $\boldsymbol{v}_t^k$ of item textual embedding as an example,

$$\begin{cases} \boldsymbol{z}^k & = \boldsymbol{W}_{intra,t}\boldsymbol{v}_t^k + \boldsymbol{b}_{intra,t}, \\ l_{intra,t}^k & = -\sum_{n=1}^{K} \tilde{z}_n^k \log \frac{\exp z_n^k}{\sum_m \exp z_m^k} \end{cases} \tag{6.3}$$

where $\boldsymbol{W}_{intra,t} \in \mathbb{R}^{K \times \frac{d}{K}}$ and $\boldsymbol{b}_{intra,t} \in \mathbb{R}^K$ are the weight matrix and bias vector of the classifier, respectively. $\boldsymbol{z}^k$ is the predicted logits and $z_n^k \in \boldsymbol{z}^k$. $\tilde{z}_n^k$ denotes the ground truth (attribute label) of chunk $\boldsymbol{v}_t^k$. For all the chunks in textual embedding $\boldsymbol{v}_t$, we have the following loss for all attribute factors in textual features:

$$l_{intra,t} = \sum_{k=1}^{K} l_{intra,t}^k. \tag{6.4}$$

Similarly, we can define the losses $l_{intra,u}$, $l_{intra,i}$ and $l_{intra,v}$ to encourage the features of each attribute factor to be concentrated in the corresponding chunk of user ID, item ID and visual feature, respectively. Finally, the total loss for the intra-

modality disentanglement is formulated as:

$$l_{intra} = l_{intra,u} + l_{intra,i} + l_{intra,t} + l_{intra,v}. \tag{6.5}$$

Inter-Modality Disentanglement

In addition to separately disentangling each modality, a serious challenge in disentangling representation learning for multi-modal features is handling the inter-relationship between the factors disentangled from multiple modalities. Intuitively, the chunks of the same attribute factor from different modality features should be consistent. For example, for the *brand* attribute, an item should have the same brand information in both the visual and textual features. In other words, chunks that share the same attribute across different modalities should be highly similar, while chunks that represent different attributes across modalities should be dissimilar.

To further achieve robust representation in multimodal recommendations, we disentangle attribute factors in different modality features by ensuring consistency of the same attribute across modalities. Inspired by this, we design a cross-modal contrastive loss as shown in Figure 6.1b. Specifically, for any two modality representations, such as $\boldsymbol{v}_t$ and $\boldsymbol{v}_v$, we take two chunks of the same attribute factor (*i.e.*, $(\boldsymbol{v}_t^k, \boldsymbol{v}_v^k)$) as positive pairs, and two chunks corresponding to different attributes (*i.e.*, $(\boldsymbol{v}_t^k, \boldsymbol{v}_v^{n\neq k})$, $(\boldsymbol{v}_t^{n\neq k}, \boldsymbol{v}_v^k)$) as negative pairs. After that, the cross-modal contrastive loss between textual and visual features is defined as follows:

$$\begin{cases} l_{t\to v}^k &= -\log \frac{\exp((\boldsymbol{v}_t^k \cdot \boldsymbol{v}_v^k)/\tau)}{\sum_{n=1}^K \exp((\boldsymbol{v}_t^k \cdot \boldsymbol{v}_v^n)/\tau)}, \\ l_{v\to t}^k &= -\log \frac{\exp((\boldsymbol{v}_v^k \cdot \boldsymbol{v}_t^k)/\tau)}{\sum_{n=1}^K \exp((\boldsymbol{v}_v^k \cdot \boldsymbol{v}_t^n)/\tau)}, \\ l_{t\leftrightarrow v} &= \sum_{k=1}^K l_{t\to v}^k + \sum_{k=1}^K l_{v\to t}^k, \end{cases} \tag{6.6}$$

where $\cdot$ represents the dot product and $\tau \in \mathbb{R}^+$ is a scalar temperature parameter. By applying this contrastive learning constraint to any two out of the three modalities, we can achieve disentanglement and alignment across features of various modalities:

$$l_{inter} = l_{i\leftrightarrow t} + l_{i\leftrightarrow v} + l_{t\leftrightarrow v}. \tag{6.7}$$

6.4.2 Low-level Attribute-driven Disentangled Representation Learning

Each attribute k is associated with a list of possible attribute values $(\tilde{y}_1^k, \tilde{y}_2^k, \cdots, \tilde{y}_{A_k}^k)$, where A_K is the total number of possible values for that attribute. For example, the attribute value of *popularity* can be stratified into five distinct levels: Super Popular, Popular, Moderate, Emerging, and Unknown. The representation disentanglement could benefit from the attribute values by exploiting their relationships of the same factor. Therefore, to learn more robust representations, we encourage disentangled representations based on attributes to predict specific attribute values of the item.

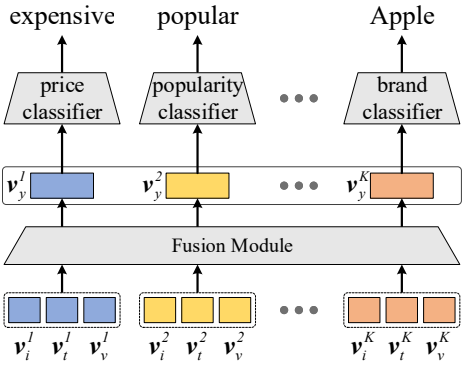


Fig. 6.2: Low-level attribute-driven disentangled representation learning module of our proposed AD-DRL. Low-level disentangled representation learning module leverages the intrinsic relationships between items sharing the same attribute value (e.g., the popularity value of different levels: *Super Popular*, *Popular*, *Moderate*, *Emerging*, and *Unknown*).

It is obvious that any single modality may not contain sufficient information for a particular attribute. For example, images may more intuitively reflect attributes such as *color* and *brand* of an item, but may not directly reflect attributes such as *price* or *popularity*. Therefore, in order to comprehensively and accurately depict the attribute values of the item, we first integrate the features from all modalities together. To achieve this, we apply a multimodal attention mechanism to measure the emphasis of different modalities on different attributes. Specifically, for the k-th attribute, the attention weights assigned to different modalities are estimated by a two-layer neural network:

$$\begin{cases} \hat{\boldsymbol{a}}^k = \boldsymbol{W}_{a2} \tanh(\boldsymbol{W}_{a1}(\boldsymbol{v}_i^k + \boldsymbol{v}_t^k + \boldsymbol{v}_v^k) + \boldsymbol{b}_a), \\ \boldsymbol{a}^k = Softmax(\hat{\boldsymbol{a}}^k), \end{cases} \tag{6.8}$$

where $\boldsymbol{W}_{a1} \in \mathbb{R}^{3\times\frac{d}{K}}$ and $\boldsymbol{W}_{a2} \in \mathbb{R}^{3\times3}$ are the weight matrices corresponding to the first and second layers of the neural network, respectively. $\boldsymbol{b}_a$ denotes the bias vector and tanh is the activation function. $Softmax$ is adopted to normalize $\hat{\boldsymbol{a}}^k$ to a probability distribution.

Thereafter, we obtain the final representation of the k-th attribute factor for the item as follows:

$$\boldsymbol{v}_y^k = a_i^k \cdot \boldsymbol{v}_i^k + a_t^k \cdot \boldsymbol{v}_t^k + a_v^k \cdot \boldsymbol{v}_v^k, \tag{6.9}$$

where a_i^k, a_t^k, $a_v^k \in \boldsymbol{a}^k$ are the attention weights of item ID embedding, textual feature and visual feature, respectively.

Similar to the disentangling at the intra-modality disentanglement in Section 6.4.1, we aim to achieve disentangled representation learning at the low level through

attribute value prediction via a classification layer (as shown in Figure 6.2):

$$\begin{cases} \boldsymbol{y}^k & = \boldsymbol{W}_k \boldsymbol{v}_y^k + \boldsymbol{b}_k, \\ l_{low}^k & = -\sum_{i=1}^{A_K} \tilde{y}_i^k \log \frac{\exp y_i^k}{\sum_j \exp y_j^k}, \\ l_{low} & = \sum_{k=1}^{K} l_{low}^k \end{cases} \tag{6.10}$$

where $\boldsymbol{W}_k \in \mathbb{R}^{A_K \times \frac{d}{K}}$ and $\boldsymbol{b}_k \in \mathbb{R}^{A_K}$ are the weight matrices and bias vector of the classifier, respectively. We supervise the training of each attribute subspace via independent attribute classification tasks defined in the form of a cross-entropy loss.

6.4.3 Prediction and Model Training

Prediction

So far we have discussed how to obtain attribute-driven disentangled representations $\boldsymbol{v}_u = (\boldsymbol{v}_u^1, \boldsymbol{v}_u^2, \cdots, \boldsymbol{v}_u^K)$ and $\boldsymbol{v}_y = (\boldsymbol{v}_y^1, \boldsymbol{v}_y^2, \cdots, \boldsymbol{v}_y^K)$ for the user u and item i, respectively. By assigning each disentangled representation with an attribute as described above, we are able to predict users' preferences at the attribute level. This enhances the interpretability and controllability of our model. Specifically, to estimate a user u's preference for an item i, it is crucial to consider her preference for each attribute of the item. To achieve this, we first compute the user's preference score for each attribute of the item, and subsequently, integrate the scores of each attribute to estimate the user's overall preference for the item:

$$\begin{cases} s_{u,i,k} = \sigma(\boldsymbol{v}_u^k \cdot \boldsymbol{v}_y^k), \\ s_{u,i} = \sum_{k=1}^{K} s_{u,i,k}, \end{cases} \tag{6.11}$$

where $\cdot$ symbolizes the dot product and $\sigma(\cdot)$ denotes the activation function. In this work, we use a softplus function to ensure the resultant score $s_{u,i,k}$ is positive. $s_{u,i,k}$ denotes the user u's preference score for the attribute k of the item i.

Training Protocol

Based on the predicted preference score above, we recommend a list of top-n ranked items that match the target user's preferences. The Bayesian Personalized Ranking (BPR) loss function [134] is employed to optimize the model parameters $\boldsymbol{\Theta}$,

$$\mathcal{L}_{BPR} = \sum_{(u,i_+,i_-)\in\mathcal{D}} -\log \phi(s_{u,i_+} - s_{u,i_-}) + \lambda \|\boldsymbol{\Theta}\|_2^2, \tag{6.12}$$

where λ is the coefficient controlling L_2 regularization; $\mathcal{D}$ denotes the training set; i_+ and i_- are the observed and unobserved items in the interaction records of user u,

respectively. Overall, the total loss of AD-DRL is formulated as,

$$\mathcal{L} = \mathcal{L}_{BPR} + \alpha \sum_{(u,i)\in\mathcal{D}} l_{intra} + \beta \sum_{(u,i)\in\mathcal{D}} l_{inter} + \gamma \sum_{(u,i)\in\mathcal{D}} l_{low}, \tag{6.13}$$

where α, β and γ are the hyperparameters to control the weight of three disentanglement modules.

6.5 Experiments

We conducted extensive experiments on three benchmark datasets to validate the effectiveness of the proposed method. In particular, the experiments are mainly performed to address the following research questions:

- **RQ1:** How does AD-DRL perform compared with existing state-of-the-art models for recommendation?
- **RQ2:** Can AD-DRL effectively learn disentangled representations by utilizing attributes?
- **RQ3:** How does our proposed AD-DRL model perform in terms of controllability and interpretability?
- **RQ4:** How do the critical components of AD-DRL contribute to the overall performance of AD-DRL on different datasets?

6.5.1 Experimental Setup

Table 6.1: Basic statistics of the three datasets. "#" denotes the number of statistical values.

Dataset	#user	#item	#interaction	sparsity	#price	#popularity	#brand	#category
Baby	12,637	18,646	121,651	99.95%	5	5	663	1
Toys Games	18,748	30,420	161,653	99.97%	5	5	1,288	19
Sports	21,400	36,224	209,944	99.97%	5	5	2,081	18

In the following, we first provide the basic information of the evaluated benchmark datasets. We then introduce the adopted evaluation metric, followed by the key baselines used in this work, and we end this section with some implementation details.

Datasets

We use the widely adopted real-world recommendation dataset, the Amazon review dataset[1] [121], for evaluation in our experiments. Apart from user-item interaction data, this dataset also includes multimodal information (*i.e.*, reviews and images) and various attributes (*i.e.*, *price*, *brand*, etc.) of the items on 24 product categories. Three product categories from this dataset are used in the evaluation: Baby, Toys Games and Sports. Following [99], for all datasets, unpopular items and inactive users are filtered out to ensure that all the items and users have at least 5 interaction records. The basic statistics of the three datasets are shown in Table 6.1.

In our setting, in addition to the user-item interaction data and multimodal information of items, multiple attributes and their associated attribute values of items are required to guide the disentangled representation learning process. Specifically, we utilize four typical attributes (*i.e.*, *price*, *popularity* [2], *brand* and *category*), and the attribute values for each attribute are compiled based on the metadata provided by the Amazon dataset: for *brand* and *category* values, we directly used the values provided by Amazon; for *price* and *popularity*, following the methods used by CoHHN [210], we discretize them into five levels according to their numerical values. Table 6.1 shows the specific statistical information for each attribute in our experiments. It is worth noting that our method can include a wide range of attributes that objectively reflect the character of an item, beyond the four attributes mentioned.

Baselines

We compared our method with the state-of-the-art methods, including both the CF-based methods (*i.e.*, NeuMF [57], CML [63],NGCF [173] and DGCF [174]) and Multimodal CF-based methods (JRL [211], MMGCN [186], MAML [101], GRCN [185], DMRL [99] and BM3 [218]). Each type of method includes a disentangled representation learning model (*i.e.*, DGCF and DMRL) for comparison. In addition to the methods mentioned above, we also created a variant of our model, denoted as AD-DRL$_{ID}$, which excludes the utilization of multimodal information, thus facilitating a fair comparison with CF-based models.

Evaluation Metrics

For each dataset, we randomly split the interactions from each user into training and testing sets with an 8 : 2 ratio. From the training set, 10% of interactions are randomly chosen as a validation set for tuning hyperparameters. We evaluate performance on the top-n recommendation task, aiming to recommend the top-n items, using Recall@n and NDCG@n as metrics for accuracy, with n defaulting to 20.

[1] http://jmcauley.ucsd.edu/data/amazon.

[2] Popularity is calculated by counting the number of times each item is purchased.

Parameter Settings

The Pytorch toolkit [132] is utilized to implement our models. To ensure fairness, all methods are optimized using the Adam optimizer [77], with a default learning rate of 0.0001 and batch size of 1024. We fixed the embedding size of each factor to 32 for AD-DRL and its variants on all datasets. The Xavier initializer [190] is used to initialize the model parameters. The α, β, γ and L_2 regularization coefficient are searched in $\{1e^{-3}, 5e^{-3}, 1e^{-2}, \cdots, 5e^{+0}, 1e^{+1}\}$. The number of negative examples is searched in $\{2, 4, 8\}$. Besides, model parameters are saved every five epochs. Early stopping strategy [173] is performed, *i.e.*, premature stopping if Recall@20 does not increase for 50 successive epochs. Our codes and datasets are released to facilitate the replication of our experiments [3].

6.5.2 Performance Comparison (RQ1)

We summarize the overall performance comparison results in Table 6.2. The methods in the first block solely use the user-item interactions, while the methods in the second block also exploit textual and visual information along with the user-item interactions. From this table, we can have the following observations:

- NeuMF, CML, NGCF, DGCF and AD-DRL$_{ID}$ are methods trained exclusively on user-item interactions. Among them, NeuMF can achieve promising performance compared to conventional MF-based methods [57]. This is attributed to the capacity of deep neural networks to model the non-linear interactions between users and items. Benefiting from the high-order information, both NGCF and DGCF accomplish state-of-the-art results. DGCF, in particular, performs better than NGCF by characterizing diverse user intents through the disentangled representation technique. This demonstrates the effectiveness of disentangled representation learning, particularly in improving the robustness of user and item representations. Furthermore, although both DGCF and AD-DRL$_{ID}$ employ disentangled representation learning, the performance of AD-DRL$_{ID}$ is significantly better than that of DGCF. This highlights the superiority of our attribute-driven disentangled representation learning approach.
- In general, the multimodal recommendation methods perform better than those only using user-item interactions, demonstrating the effectiveness of leveraging multimodal information on learning user and item representations. Although the simple neural network structure results in lower performance of JRL compared to the graph-based methods (NDCG and DGCF), incorporating rich multi-modal information enables it to outperform NeuMF. By exploiting user-item interactions to guide the representation learning in different modalities, MMGCN outperforms NGCF on all the datasets. MAML outperforms NGCF by modeling users' diverse preferences by using multimodal features of items. GRCN sur-

[3] https://github.com/SDLZY/AD-DRL.

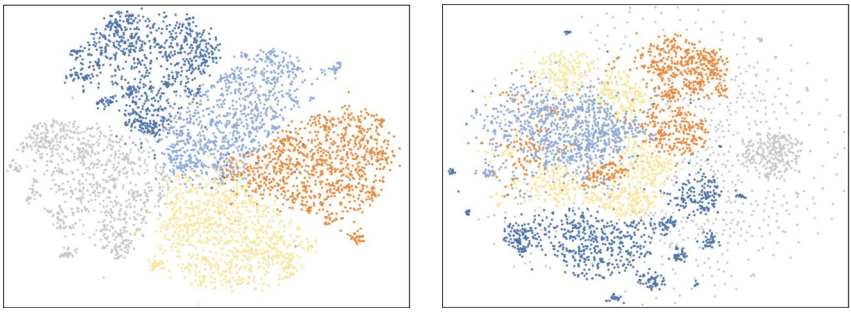
Table 6.2: Performance comparison of different recommendation methods over three datasets. The methods are categorized into two categories (i.e., blocks): 1) methods in the 1st block are the ones relying only on the user-item interaction information without any side information, and 2) methods in the 2nd block are the ones that consider multimodal information. The best results are highlighted in bold.

Datasets	Baby		Toys Games		Sports	
Metrics	Recall	NDCG	Recall	NDCG	Recall	NDCG
NeuMF	0.0502	0.0224	0.0253	0.0128	0.0330	0.0157
CML	0.0674	0.0444	0.1227	0.1160	0.0982	0.0640
NGCF	0.0694	0.0313	0.0970	0.0587	0.0707	0.0337
DGCF	0.0788	0.0465	0.1262	0.1085	0.1026	0.0629
AD-DRL$_{ID}$	0.0852	0.0529	0.1517	0.1429	0.1120	0.0722
JRL	0.0579	0.0266	0.0472	0.0413	0.0368	0.0214
MMGCN	0.0814	0.0496	0.1171	0.1065	0.0913	0.0572
MAML	0.0867	0.0521	0.1183	0.1117	0.1029	0.0676
GRCN	0.0883	0.0541	0.1336	0.1236	0.1065	0.0693
DMRL	0.0906	0.0561	0.1434	0.1331	0.1111	0.0711
BM3	0.0911	0.0424	0.1147	0.0683	0.1121	0.0536
AD-DRL	**0.0968***	**0.0588***	**0.1524***	**0.1435***	**0.1200***	**0.0756***
Improv.	6.84%	4.81%	6.28%	7.81%	8.01%	6.33%

The symbol * denotes that the improvement is significant with $p - value < 0.05$ based on a two-tailed paired t-test.

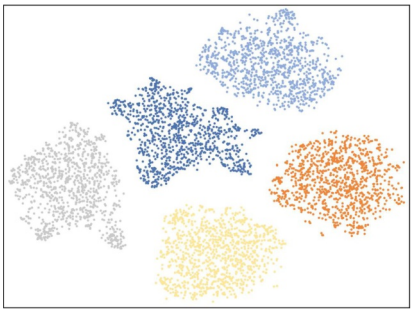
passes MAML by employing modality features to discover and prune potential false-positive edges on the user-item interaction graph. DMRL performs better than the other baselines over almost all the datasets. It should be credited to its capability to capture the different contributions of features from different modalities for each disentangled factor.

- DGCF and DMRL, two models based on disentangled representation learning, outperform other single-modal and multi-modal recommendation models across all three datasets. This highlights the effectiveness of disentangled representation learning, particularly in improving the robustness of representations and characterizing diverse user intents.
- Furthermore, our proposed AD-DRL model consistently outperforms all baselines over all three datasets by a large margin. We credit this to the joint effects of the following two aspects. Firstly, robust user and item representations can be derived using attributes at different levels to guide the disentangled representation learning process. Secondly, incorporating the additional attribute information into representation learning can alleviate the data sparsity problem in the recommendation. Moreover, the comparison between AD-DRL and AD-DRL$_{ID}$ shows that the multimodal information, which contains more sufficient informa-

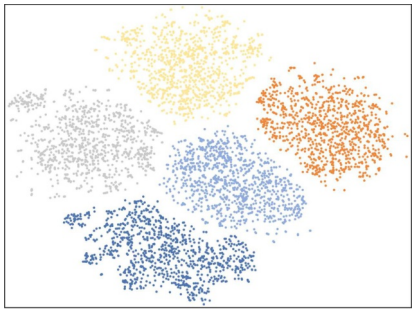


(a) Disentangled vectors (User ID)

(b) Disentangled vectors (Item ID)



(c) Disentangled vectors (Review)



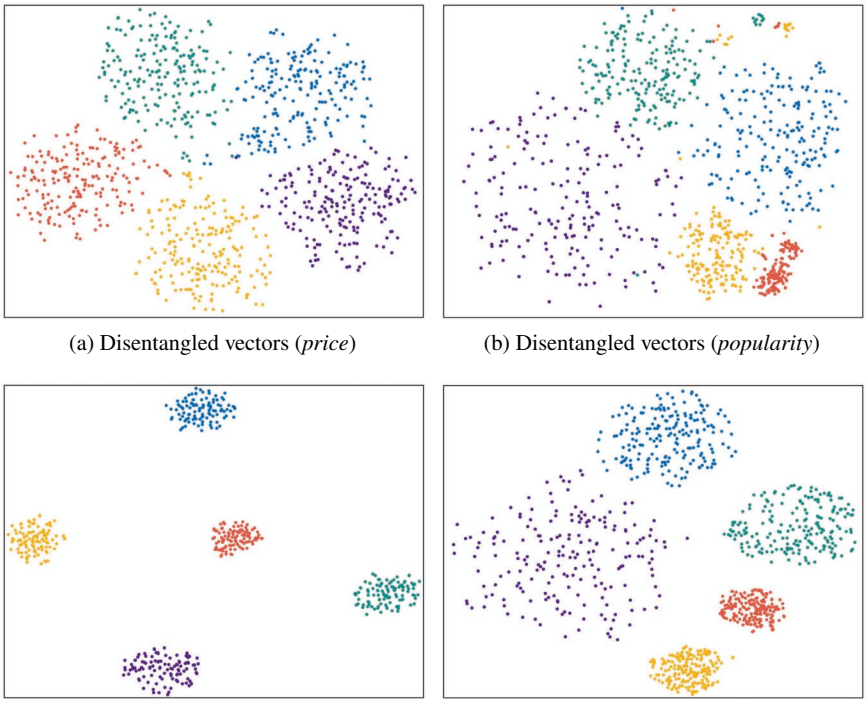
(d) Disentangled vectors (Image)

Fig. 6.3: Visualization of disentangled vectors from ID embeddings and different modalities, with distinct colors indicating different attributes: yellow for *price*, orange for *popularity*, grey for *brand*, dark blue for *category*, and light blue for *others*.

tion for various attributes, can enhance the effect of attribute-driven disentangled representation learning.

6.5.3 Visualization of Disentangled Representations (RQ2)

AD-DRL conducts disentangled representation learning for user and item representations at two levels of attribute granularity: high and low. To further confirm the effectiveness of disentangled representation learning by AD-DRL, we use t-SNE [120] to cluster and visualize the disentangled vectors of users and items from the Sports dataset at each granularity level. t-SNE (t-Distributed Stochastic Neighbor Embedding) is a popular dimensionality reduction technique primarily used for visualizing high-dimensional data in a lower-dimensional space (typically 2D or 3D). It is especially effective for exploring complex datasets, such as those in machine



(a) Disentangled vectors (*price*) (b) Disentangled vectors (*popularity*)

(c) Disentangled vectors (*brand*) (d) Disentangled vectors (*category*)

Fig. 6.4: Visualization of the disentangled vectors corresponding to different attributes, where different colors represent different attribute values.

learning or NLP, where the relationships between data points are difficult to discern in high-dimensional spaces.

High-Level Attribute-driven Disentanglement

Figure 6.3 displays the disentangled vectors processed by the high-level attribute-driven disentanglement module for each modality feature, where dots of the same color denote the vectors corresponding to the same attribute. It can be observed that our model effectively distinguishes representation vectors corresponding to different attributes for both users and items across various modalities, demonstrating the effectiveness of our model in disentangling at the attribute level. Such disentangled representations can better capture user preferences and item characteristics toward different attributes.

Low-Level Attribute-driven Disentanglement

To verify the effectiveness of low-level (attribute value-level) disentanglement, we visualize the representations of each attribute factor (*i.e.*, $\boldsymbol{v}_y^k$ in Equation 6.9) in Figure 6.4. The dots of different colors represent different attribute values[4]. As observed, the representations of different attribute values learned from AD-DRL are well separated and the representations of the same attribute value are concentrated. These results demonstrate that AD-DRL can achieve finer-grained disentanglement at the attribute value level, allowing AD-DRL to capture better user preferences.

6.5.4 Study of Interpretability and Controllability (RQ3)

Interpretability

To gain a deeper insight into the interpretability of our model, in this section, we provide some qualitative examples in Figure 6.5. As an illustrative example, we randomly sample two users ($u18629$ and $u6805$) from the Sports dataset who purchased the same items ($i2099$ and $i3460$). Since we have disentangled the representation of users and items based on the attributes, we can analyze how each attribute contributes to recommendation results based on Equation 6.11. This greatly enhances the interpretability of our model. For example, *price* contributes most to the interaction ($u10826$, $i2099$). This suggests that $u10826$ prefer $i2099$ due to its *price*. However, $u6805$ is more likely to purchase the $i2099$ because of her preference for its *brand*. Such observation indicates that different users have diverse preferences for the same product. In addition, we can also see that user preferences exhibit consistency. For example, $u6905$ values the *brand* of the product more than its *popularity* or *price*, which is different from $u10826$.

Controllability

We conducted experiments to evaluate the controllability of AD-DRL by manipulating user preferences for certain attributes. Specifically, we change the user u's preference score for a specific attribute a in Equation 6.11 to assess if such change yields the anticipated variation in recommendation results. We modify the formula for $s_{u,i}$ in Equation 6.11 as follows:

$$s_{u,i} = \xi * s_{u,i,a} + \sum_{k \neq a} s_{u,i,k}, \tag{6.14}$$

[4] For *brand* and *category*, since there are too many attribute values in the dataset, making it difficult to display all of them in Figure 6.4. Therefore, for these two attributes, we only selected the top 5 attribute values with the most corresponding items in the dataset for demonstration.

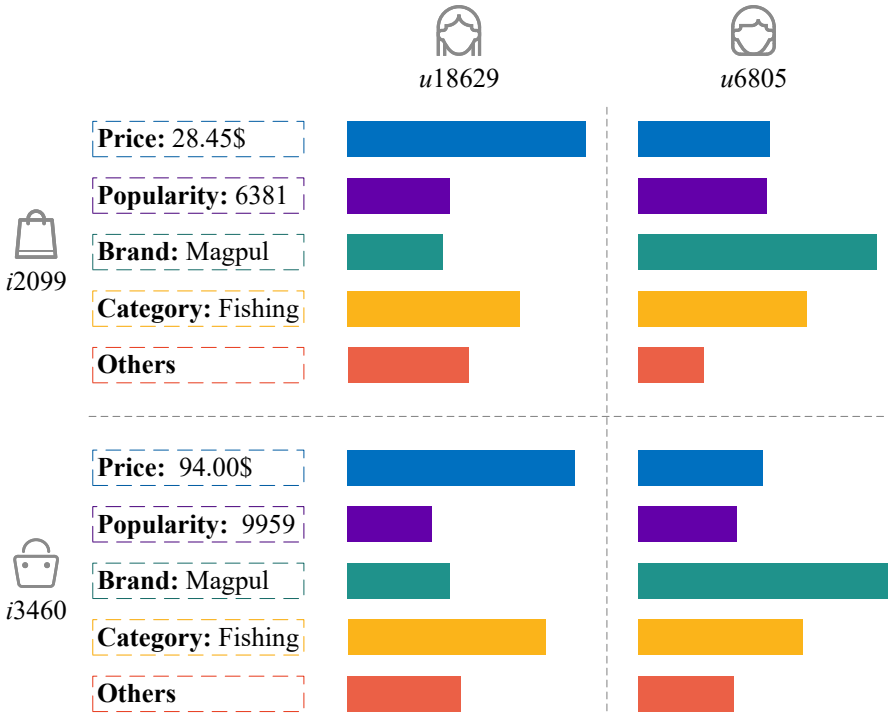
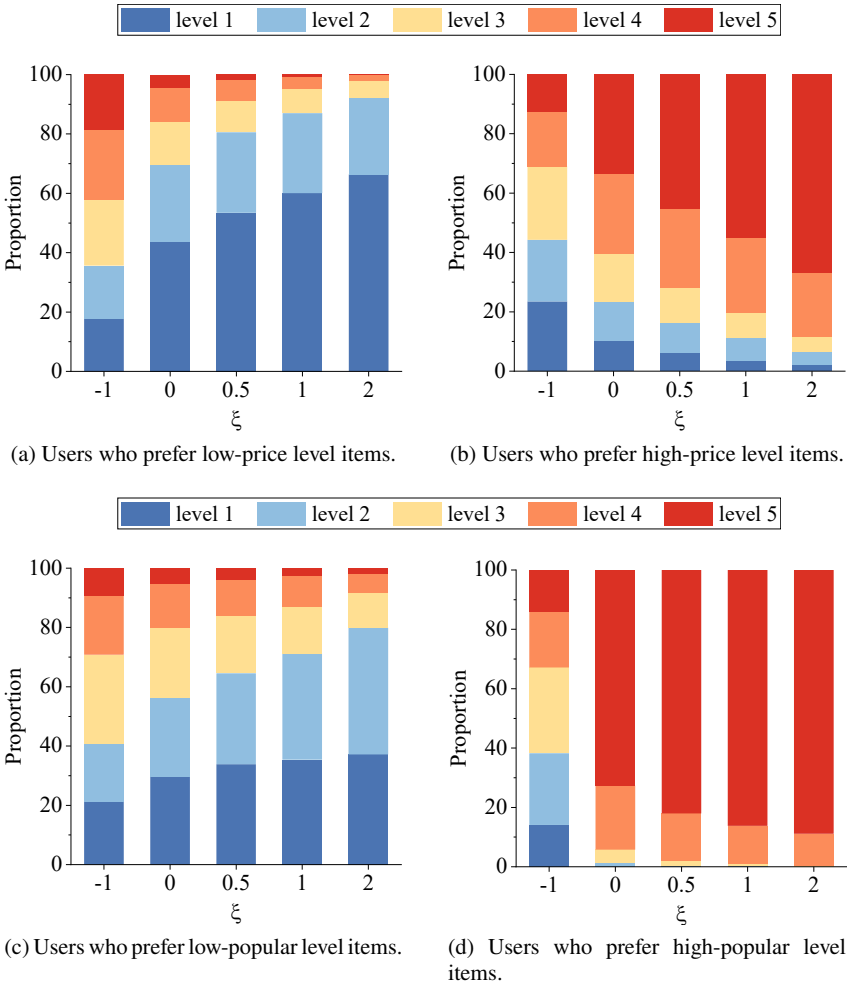


Fig. 6.5: The preference scores of two users ($u18629$ and $u6805$) for different attributes of two items ($i2099$ and $i3460$).

where ξ represents the scaling factor for $s_{u,i,a}$, used to adjust the impact of attribute a on $s_{u,i}$. For a specific attribute (in this section, we selected *price* and *popularity* for study), we systematically vary ξ through values of 2, 1, 0.5, 0, and -1. This process allows us to meticulously examine the resultant shifts in our method's recommendation outputs, with findings illustrated in Figure 6.6.

We conduct a detailed analysis using the *price* attribute as an example. As shown in Figure 6.6a, we select 100 users from *Baby* dataset who always purchase inexpensive items based on their purchase records. Various colors in Figure 6.6a denote distinct price levels, showcasing the distribution of items recommended by AD-DRL across these levels for a given value of ξ. By comparing the recommendations made by AD-DRL under different values of ξ, we have the following observations:

- When $\xi = 1$, that is, using the original AD-DRL method, it can be seen that AD-DRL is capable of capturing the preference of these users for inexpensive items and recommends more affordable items to them.
- When $\xi = 2$, meaning we have increased the impact of *price* on the user preference score, it can be observed that AD-DRL tends to recommend more inexpensive items to these users.



(a) Users who prefer low-price level items.

(b) Users who prefer high-price level items.

(c) Users who prefer low-popular level items.

(d) Users who prefer high-popular level items.

Fig. 6.6: The proportion of items with different attribute values within the AD-DRL's recommendations when ξ takes different values in Equation 6.14.

- When $\xi = 0.5$ or 0, meaning we reduce or even eliminate the impact of *price* on the user preference score, we find that the recommended items from AD-DRL are more diverse in terms of *price*. It is worth noting that even when $\xi = 0$, the items recommended by AD-DRL still tend to be inexpensive. This could be attributed to the retained chunk associated with the *brand*, and there are correlations between *brand* and *price*. For example, some high-end brands offer products at a premium price, while others provide more affordable options.

Table 6.3: Ablation study of our proposed AD-DRL method over three datasets. The best results are highlighted in bold.

Datasets	Baby		Toys Games		Sports	
Metrics	Recall	NDCG	Recall	NDCG	Recall	NDCG
AD-DRL$_{w/o\ \text{disentangling}}$	0.0888	0.0550	0.1452	0.1375	0.1152	0.0741
AD-DRL$_{w/o\ \text{intra}}$	0.0925	0.0567	0.1492	0.1397	0.1164	0.0750
AD-DRL$_{w/o\ \text{inter}}$	0.0959	0.0585	0.1517	0.1429	0.1196	0.0754
AD-DRL$_{w/o\ \text{high}}$	0.0903	0.0564	0.1492	0.1391	0.1160	0.0748
AD-DRL$_{w/o\ \text{low}}$	0.0916	0.0553	0.1492	0.1394	0.1177	0.0755
AD-DRL	**0.0968**	**0.0588**	**0.1524**	**0.1435**	**0.1200**	**0.0756**

- More interestingly, when $\xi = -1$, meaning we have AD-DRL recommend items that are opposite to the users' *price* preferences, it can be observed that AD-DRL indeed recommends some more expensive items.

The above results reveal that adjusting the value of ξ enables us to align AD-DRL's recommendation results with our expectations, thereby illustrating the controllability of AD-DRL. Similar results can be seen in another group of users who prefer high-price level items.

6.5.5 Ablation Study (RQ4)

We conduct detailed experiments to validate the effectiveness of high-level and low-level attribute-driven disentangled representation learning modules. Further, to deepen our insight into the high-level attribute-driven disentangled representation learning module, we also investigate the efficacy of the intra- and inter-modality disentanglement modules. Specifically, we consider the following variants of AD-DRL for evaluation:

- AD-DRL$_{w/o\ \text{disentangling}}$: we do not perform disentangled representation learning during the training process;
- AD-DRL$_{w/o\ \text{intra}}$: we only remove the intra-modality disentanglement module;
- AD-DRL$_{w/o\ \text{inter}}$: we only remove the inter-modality disentanglement module;
- AD-DRL$_{w/o\ \text{high}}$: we remove the high-level attribute-driven disentangled representation learning module;
- AD-DRL$_{w/o\ \text{low}}$: we remove the low-level attribute-driven disentangled representation learning module.

Table 6.3 shows the experimental results for all variants. From the results, we have the following observations:

- Using either high-level or low-level disentangled representation learning independently can significantly improve the performance. This indicates that both disentangled representation learning within modalities and between modalities have a positive effect.
- The AD-DRL combines high-level and low-level disentangled representation learning modules, resulting in significantly improved performance compared to the devised two variants. This demonstrates the necessity and rationality of combining the attribute and attribute value levels.
- Delving deeper into the high-level disentangled representation learning module, removing the intra-modality disentanglement module results in a greater decrease in model performance compared to removing the inter-modality disentanglement module. This indicates that intra-modality disentanglement is the fundamental aspect of disentangled representation learning, while inter-modality disentanglement can further enhance model performance.

6.6 Conclusion

In this chapter, we highlight the limitations of existing disentangled representation learning techniques in recommender systems. The current methods disentangle the underlying factors behind user-item interactions in an unsupervised manner, leading to limited interpretability and controllability. To overcome this limitation, we propose an attribute-driven disentangled representation learning method, termed AD-DRL, which disentangles attribute factors in various multimodal features. More specifically, the proposed AD-DRL enables disentangling within each modality feature and ensures consistency of representations across modalities at the attribute level. Moreover, it leverages the intrinsic relationships between items sharing the same attribute value. Experimental results demonstrate the superiority of our proposed AD-DRL and showcase its capability in terms of interpretability and controllability.

Chapter 7
Research Frontiers

This book offers a comprehensive exploration of multimodal learning techniques in recommendation systems, focusing on five critical challenges: noise and redundant features, user diverse preference, user modality preference, dynamic multimodal fusion, and the interpretability and controllability of recommendations. These challenges represent significant barriers to effectively leveraging multimodal data in recommendation systems. To overcome these obstacles, the book introduces a set of innovative, targeted approaches designed to enhance the utility and performance of multimodal recommendations.

Specifically, we introduce a semantic-guided feature distillation approach to tailor generic multimodal features extracted from raw data by pre-trained models for recommendation. This establishes a solid foundation for the efficient utilization of multimodal data. Next, we propose a multimodal attention metric learning method that combines attention mechanisms with the metric-based learning method. This enables the modeling of users' diverse preferences on different aspects of items while satisfying the triangle inequality property. To capture user modality preferences, we develop a disentangled multimodal representation learning approach, which learns user attention towards item factors in different modalities. This allows the recommendation system to effectively leverage multimodal information by considering user preference toward various factors in each modality. To capture the various relationships among multimodal information of different items, we develop a meta-learning-based multimodal fusion framework, which dynamically assigns parameters to the multimodal fusion function for each item during its representation learning. Finally, to enhance both the interpretability and controllability of recommendation outcomes, we propose an attribute-driven disentangled representation learning approach, utilizing item attributes from multimodal data to facilitate explanation and control over recommendation results, ultimately improving user satisfaction. To foster further research, we have also open-sourced the relevant datasets, codes, and parameter settings associated with this book.

The proposed solutions systematically address the five key challenges in multimodal recommendation, thereby significantly enhancing the effectiveness of multimodal data in driving more accurate and personalized recommendations. By tackling

F. Liu et al., *Multimodal Learning toward Recommendation*,
https://doi.org/10.1007/978-3-031-83188-1_7

these challenges, the contributions of this book demonstrate substantial promise for broad applications across a variety of internet platforms, including e-commerce, social media, and beyond. These advancements are poised to benefit a diverse array of stakeholders, including merchants, content creators, and end-users. For merchants, the adoption of multimodal learning techniques can help alleviate the over-reliance on traditional recommendation systems, which primarily depend on user-item interaction records. This shift reduces the impact of data sparsity and cold start problems—two common challenges that often hinder the performance of conventional recommendation algorithms. By leveraging multimodal data, such as images, text, and video, merchants can provide more contextually relevant and engaging recommendations to users, thereby improving user experience. Ultimately, these improvements contribute to higher conversion rates, increased customer satisfaction, and better sales outcomes. Additionally, the effective integration of multimodal information can drive significant improvements in user satisfaction, retention, and operational efficiency. By leveraging diverse data sources, organizations can optimize inventory management and better allocate resources, ensuring that offerings align more closely with user demands. For content creators, multimodal recommendation systems provide a deeper understanding of the various types of content, allowing for greater visibility and more precise targeting in content distribution. This enables creators to reach the right audiences with more relevant content, enhancing their impact and engagement. For end-users, multimodal recommendation systems offer a richer, fine-grained understanding of their interests and preferences, leading to personalized and contextually relevant suggestions. Moreover, the incorporation of multimodal information enhances the transparency of recommendations by providing rational explanations for why certain items are suggested. This transparency empowers users with greater control over their content experience, fostering trust and satisfaction, and ultimately contributing to a more engaging and rewarding interaction with the platform.

Despite the extensive discussion and proposed solutions for the key technical challenges in multimodal learning for recommendation systems, we recognize that this research area is still in its early stages, with considerable opportunities for further exploration. In this regard, the book outlines several promising directions and challenges, aiming to inspire and guide future research in this evolving domain.

7.1 Multimodal Learning with Large Language Models

With the rapid rise of Large Language Models (LLM) like ChatGPT [1], significant success has been achieved in fields such as natural language processing [12, 180, 179] and computer vision [107, 17, 163]. Relevant work suggests that large language models have great potential for application in the recommendation domain, and an increasing number of researchers are focusing on how to apply these models in rec-

[1] https://chatgpt.com/

ommendation systems [98, 181, 76]. Traditional recommendation systems typically rely on methods like collaborative filtering or content similarity, which have certain limitations in terms of data sparsity and understanding complex contexts. In contrast, large language models can handle massive amounts of natural language data and possess strong semantic understanding and generative capabilities [111, 96], giving them significant advantages in user interest modeling, complex intent comprehension, and diverse recommendations. Through a deep understanding of user behavior, large language models can provide not only personalized recommendations but also explore latent user needs, thereby enhancing the user experience and demonstrating significant value in improving recommendation accuracy, expanding recommendation scenarios, and mitigating the cold start problem. This combination of natural language processing and recommendation systems offers new possibilities for creating more intelligent and user-friendly recommendation systems.

Preliminary efforts have already begun to explore and achieve success in this area [154, 197, 207, 7, 52]. For example, LLaRA [97] introduce a innovative hybrid representation learning approach for items in LLM input prompts by combining ID-based item embeddings from traditional recommendation systems with textual item features. This integration capitalizes on the complementary strengths of traditional recommenders, which encode user behavioral data, and LLMs, which possess extensive world knowledge about items. For instance, to mitigate the time and computational overhead associated with processing lengthy texts in recommendation scenarios, LLM-TRSR [216] employs an LLM to summarize users' historical interactions, thereby streamlining the information and enabling more efficient LLM-based recommendations.

Despite the progress achieved by existing LLM-based methods, effectively incorporating multimodal information remains a significant and complex challenge. Specifically, the integration of multimodal learning with LLMs faces several key obstacles: 1) Modality gap. LLMs are primarily designed with text as the core input modality, and their ability to process visual information is relatively limited. While some studies have attempted to convert images into text for input into LLMs, this approach often leads to inevitable information loss [116, 201]. 2) Integration of multimodal and collaborative filtering information [76]. Combining collaborative filtering with large models has long been a challenge in recommender systems, and the introduction of multimodal data further complicates this issue. Effectively integrating multimodal information with collaborative filtering remains a pressing challenge that needs to be addressed. 3) Controllability of Recommendation Results. Compared to traditional recommendation models, the output of LLMs is difficult to control and even slight modifications to prompts can cause significant changes in recommendation results. Therefore, when introducing multimodal information, achieving controllable outputs from LLMs is a long-standing key challenge.

To address the above issues, we plan to explore the following approaches to enhance the performance of LLM-based recommender systems through multimodal learning: 1) We aim to develop a new paradigm that enables lossless extraction of multimodal features to improve the utilization efficiency of multimodal information in recommender systems. 2) We intend to explore the synergistic mecha-

nisms between multimodal information and collaborative filtering information to build recommender systems that leverage the advantages of both. 3) We will utilize Retrieval-Augmented Generation (RAG) technology [88, 44, 4] to optimize how LLMs acquire and use of external information, thereby enhancing the controllability of their outputs. Through these approaches, we hope to provide new insights and technical support for the development of multimodal recommender systems based on LLMs.

7.2 Multimodal Learning Towards Open-world Recommendation

Traditional recommendation models are typically designed for specific scenarios or categories of items, under the assumption that the user and item sets remain known and fixed within a defined timeframe - this is referred to as closed-world recommendation. However, such a conventional paradigm falls short of meeting the dynamic and diverse requirements of real-world users, who demand personalized recommendations across various domains, categories, and changing contexts. To address these limitations, a novel research paradigm known as open-world recommendation has emerged [191, 189, 192]. This approach aims to deliver effective recommendations in scenarios where new users and items continuously appear, without any restriction to the context. Given the necessity for open-world recommendation systems to have a comprehensive perception of the real world, multimodal learning is expected to play a significant role in advancing this field.

Compared to traditional recommendation models, open-world recommendation presents notable advantages and promising applications: (1) **Broader recommendation scope.** Open-world recommendation can effectively address multiple scenarios concurrently, whereas traditional models are typically designed to handle one scenario at a time; (2) **Enhanced adaptability and flexibility.** Open-world recommendation is capable of dynamically adapting to the introduction of new users and items, which makes it particularly well-suited for rapidly evolving contexts, such as e-commerce, news platforms, and social media. This dynamic nature allows the system to respond efficiently to changing user preferences and item availability; (3) **Mitigated cold-start problem.** By employing mechanisms to quickly integrate new data, open-world recommendation systems can significantly alleviate the cold-start issue [140], enabling more rapid and personalized recommendations for newly introduced users or items.

Despite the evident advantages, open-world recommendation presents several challenges that have not yet been thoroughly addressed, especially in terms of leveraging multimodal information. The primary challenges are as follows: (1) **Data sparsity.** The long-tail distribution problem, which plagues traditional recommendation scenarios, becomes even more pronounced in open-world settings. Effectively integrating multimodal information—such as text, images, and audio—presents a significant opportunity to mitigate data sparsity, but also poses unique challenges; (2) **Real-time model updates and efficiency.** Open-world recommendation requires

the system to quickly respond to frequent data changes, necessitating high standards of efficiency and real-time performance. Consequently, optimizing the efficiency of multimodal learning processes is imperative to meet these demands; (3) **Acquisition and extraction of external information.** Open-world recommendation relies on the collection of useful external multimodal data while minimizing the influence of noise and redundancy. This requirement sets high expectations for feature extraction techniques, as the system must effectively distinguish between valuable information and irrelevant data.

To overcome these challenges, we plan to explore the enhancement of open-world recommendation through multimodal learning from the following perspectives: (1) **Mitigating cold-start issues for new items.** We intend to leverage large multimodal models to mine the textual and visual information associated with items, utilizing the inherent knowledge embedded in these models to enable content-based recommendation without reliance on historical user-item interactions; (2) **Enabling real-time updates with RAG.** We aim to implement RAG techniques [143, 133] to acquire and integrate relevant multimodal information in real time, thus keeping the recommendation system up-to-date with external world knowledge. By combining these strategies, we seek to enhance the robustness and adaptability of open-world recommendation systems, ultimately providing more accurate and context-aware recommendations.

7.3 Privacy-Preserving Techniques for Multimodal Recommendation

As the recommendation methods are heavily data-driven, the more data it uses, the better is the obtained recommendation [57]. However, such users' data is often highly private for numerous personal, social, and financial reasons [8]. The risks concerning violating individuals' privacy present a significant impediment to storing or using the users' private data in an insecure environment [42, 164]. For instance, the user's private data may be shared with various organizations or stored in the public cloud. The integration of multimodal data in recommender systems further complicates these privacy concerns, as such data often contains rich layers of user-specific information, thereby introducing heightened privacy risks. To ensure user trust, multimodal recommender systems must prioritize the robust protection of users' personal data throughout its collection, processing, and transmission, thereby preventing the leakage or misuse of sensitive information. Additionally, strengthening user privacy protection in multimodal learning is also crucial for helping enterprises comply with data privacy laws in various countries and for building user trust in the platform, thereby further encouraging users to share data. This is of great significance for the development of recommender systems.

However, achieving privacy protection in recommender systems is not easy, especially in multimodal recommendation, where the complexity of privacy protection is more pronounced compared to traditional recommender systems based on a single

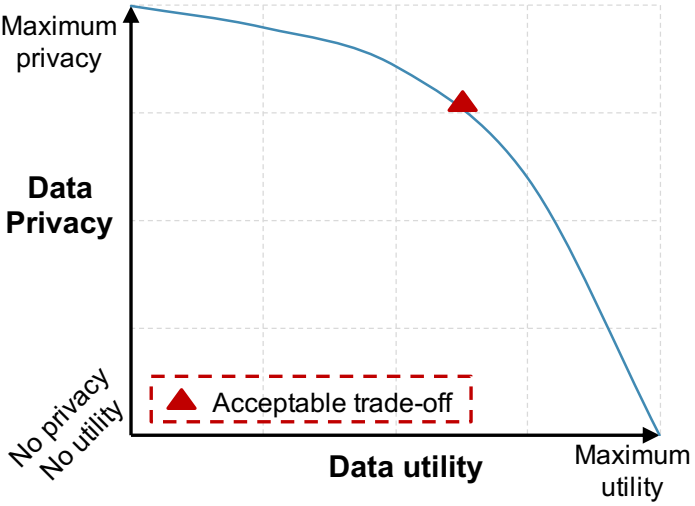


Fig. 7.1: The trade-off between data privacy and data utility.

modality (such as transaction data). This is mainly reflected in the following three aspects. First, the trade-off between recommender system performance and privacy protection is a long-standing issue. Privacy protection techniques may reduce model performance, making the already complex multimodal learning even more challenging. Generally, as shown in Fig. 7.1, the stronger the privacy protection, the greater the performance loss of the recommendation model [100]; therefore, it is necessary to find an appropriate balance between privacy protection and recommendation effectiveness. Second, some privacy protection methods (such as Federated learning [196]) often require substantial computation and communication resources, and the complexity of multimodal information may further affect the system's real-time performance and scalability. Lastly, multimodal data also faces issues such as data imbalance, inconsistent quality, missing values, and noise, which increase the difficulty of privacy protection.

To address the above challenges, we plan to conduct research from the following aspects: (1) Design user privacy-controllable data generation models based on users' preferences in different domains and for different items, to achieve users' autonomous control over the balance between recommendation effectiveness and privacy protection. (2) Develop privacy protection models capable of handling multimodal data — for example, utilizing clients' local multimodal data for model training within a federated learning framework, with the server only responsible for aggregating model parameters, thereby reducing the risk of privacy leakage. (3) Design cross-modal privacy protection methods by using one modality to protect another modality — for instance, using text data to generate privacy protection parameters related to image data.

7.4 Fairness Mitigation in Multimodal Recommendations

Ensuring fairness in recommendation systems involves minimizing biases across different user groups to guarantee equitable access to recommendation outcomes for all users [175, 95, 46, 93]. Multimodal recommendation systems, which leverage various types of data (*e.g.*, text, images, and metadata) to enhance recommendation quality, are particularly susceptible to inadvertently introducing biases due to the inherent complexity of multimodal data [142]. Consequently, certain user groups may receive recommendations that are systematically inferior to others, which poses significant challenges for fairness. For instance, consider a scenario in which one group of users (*Group A*) frequently provides reviews, including images and textual feedback, whereas another group (*Group B*) rarely provides such data. In this case, the multimodal learning algorithm tends to rely predominantly on the richer multimodal data provided by *Group A*, leading to a model biased toward *Group A*'s preferences while disregarding the needs of *Group B*. Such bias can result in disproportionately favorable recommendations for *Group A*, thereby disadvantaging *Group B*. Addressing this issue (specifically, identifying, quantifying, and mitigating such biases) is essential to fostering equitable recommendation practices and improving the overall user experience.

The pursuit of fairness in multimodal recommendation systems is, however, confronted by several challenges, and the current research in this area remains relatively nascent. One primary challenge is that the training data often contain historical biases, which may inadvertently be learned and reinforced by the model, resulting in unfair outcomes. Additionally, in multimodal data, the degree and forms of data bias may be vary across different modalities. Understanding how to mitigate these biases in a fine-grained manner across modalities while considering their unique characteristics is a complex problem. Furthermore, the process of enhancing fairness may result in a trade-off with recommendation accuracy, presenting the challenge of finding an optimal balance between these two competing objectives.

To address these challenges, we plan to propose a multifaceted approach to enhancing fairness in multimodal recommender systems, focusing on three critical levels: data, model, and result post-processing. At the data level, we will adopt data debiasing and augmentation techniques. This involves identifying and either removing or balancing biased samples before model training, alongside augmenting data for underrepresented groups to better balance data distribution and mitigate biases. At the model level, we plan to introduce adversarial training strategies that render the recommendation system insensitive to users' sensitive attributes, such as gender and race. Additionally, we propose treating fairness as an independent optimization objective, allowing for joint optimization with traditional recommendation performance metrics. Such a multi-objective optimization approach will ensure that fairness is adequately prioritized during the training process while maintaining recommendation accuracy. Finally, at the result level, we aim to design post-processing techniques that adjust the generated recommendations to ensure equitable distribution across different user groups. We will also introduce fairness evaluation metrics,

such as coverage and diversity, to assess the quality of recommendations for each user group, providing a comprehensive understanding of fairness outcomes.

Establishing fairness in recommendation systems is not solely a technical endeavor; it requires a multidisciplinary perspective to formulate ethical frameworks that underpin the design of equitable AI-driven systems. As such, beyond the technical interventions at the data, model and result levels, our future research will also incorporate insights from sociology, ethics and related disciplines to develop fairness strategies that are both comprehensive and aligned with societal norms and legal requirements.

7.5 Bias Mitigation in Multimodal Recommendations

Multimodal recommender systems, which integrate diverse types of data, present more intricate bias-related challenges compared to their unimodal counterparts. These biases include data bias (resulting from data imbalance) [43], user bias (stemming from variability in user behavior) [219], and modality bias (arising from the differential importance assigned to distinct modalities) [198]. Such biases can significantly undermine the accuracy of recommendation results, ultimately diminishing the quality of the user experience [19, 199]. Specifically, biases may cause the recommender system to favor certain types of content, thereby reducing the diversity of recommendations. Moreover, the presence of bias may lead to feedback loops, wherein existing biases are further reinforced. For instance, by continuously recommending content that aligns with a user's past preferences, the system perpetuates a lack of content diversity. Accordingly, bias mitigation in recommender systems has garnered substantial attention from the researchers in recent years.

In multimodal recommender systems, the elimination of bias presents several significant challenges. First, a significant imbalance often exists among different modalities. For instance, the volume of textual data typically far exceeds that of video data. Such imbalance can lead to an overrepresentation of certain modalities in the recommendation outcomes, resulting in unintended biases that may degrade the system's fairness and accuracy. Second, biases inherent within individual modalities impose higher demands on the fusion process in multimodal settings [114]. Current multimodal fusion approaches may further exacerbate these biases, leading to a model that is overly reliant on certain modalities during the fusion process, thereby intensifying existing biases. Moreover, the sources of bias in multimodal recommender systems are often multifaceted and deeply interwoven, making their identification a particularly challenging task [112]. Biases may stem from diverse aspects such as data collection, user preferences or model architecture, and these sources can interact in complex ways that are difficult to untangle. Specifically, biases may be concealed within intricate model structures and modality interactions, rendering them challenging to explicitly detect and mitigate. Therefore, effectively addressing bias in multimodal recommendation systems necessitates a comprehen-

sive approach that considers not only data-level interventions but also model-level adaptations to ensure balanced and fair information integration across all modalities.

To address the aforementioned challenges, we plan to propose a series of potential solutions that can serve as a foundation for future research. First, to alleviate data imbalance, techniques such as undersampling and oversampling can be utilized to balance the data across different modalities, thereby reducing the impact of modality bias. Furthermore, data augmentation methods can be employed to enhance the representation of certain modalities, thereby promoting greater uniformity in data distribution. Second, incorporating attention mechanisms into the model architecture provides a means to dynamically adjust the relative contributions of different modalities, thereby mitigating the adverse effects of over-reliance on any single modality. Such mechanisms enable the model to automatically recalibrate modality importance based on the specific context or task, facilitating more equitable and precise recommendations. Finally, analyzing user behavior data can help identify potential sources of bias, which can then be rectified through targeted interventions. For instance, diversity-promoting recommendation strategies can be implemented to encourage users to explore content beyond their established preferences, thereby mitigating preference-based biases and enhancing the overall comprehensiveness and diversity of the recommendations provided.

References

1. MARS: Memory attention-aware recommender system. IEEE (2019). DOI 10.1109/DSAA.2019.00015
2. Ahn, S., Hu, S.X., Damianou, A.C., Lawrence, N.D., Dai, Z.: Variational information distillation for knowledge transfer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9155–9163. IEEE Computer Society (2019). DOI 10.1109/CVPR.2019.00938
3. Arora, S., Liang, Y., Ma, T.: A simple but tough-to-beat baseline for sentence embeddings. In: Proceedings of the International Conference on Learning Representations, pp. 1–16. OpenReview.net (2017)
4. Asai, A., Wu, Z., Wang, Y., Sil, A., Hajishirzi, H.: Self-rag: Learning to retrieve, generate, and critique through self-reflection. In: Proceedings of the International Conference on Learning Representations. OpenReview.net (2024)
5. Bachrach, Y., Finkelstein, Y., Gilad-Bachrach, R., Katzir, L., Koenigstein, N., Nice, N., Paquet, U.: Speeding up the xbox recommender system using a euclidean transformation for inner-product spaces. In: Proceedings of the ACM Conference on Recommender Systems, pp. 257–264. ACM (2014). DOI 10.1145/2645710.2645741
6. Baltrušaitis, T., Ahuja, C., Morency, L.P.: Multimodal machine learning: A survey and taxonomy. Transactions on Pattern Analysis and Machine Intelligence **41**(2), 423–443 (2018). DOI 10.1109/TPAMI.2018.2798607
7. Bao, K., Zhang, J., Zhang, Y., Wang, W., Feng, F., He, X.: Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In: Proceedings of the ACM Conference on Recommender Systems, pp. 1007–1014. ACM (2023). DOI 10.1145/3604915.3608857
8. Barak, B., Chaudhuri, K., Dwork, C., Kale, S., McSherry, F., Talwar, K.: Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In: Proceedings of the Symposium on Principles of Database Systems, pp. 273–282. ACM (2007). DOI 10.1145/1265530.1265569
9. Bell, R.M., Koren, Y.: Lessons from the netflix prize challenge. ACM SIGKDD Explorations Newsletter **9**(2), 75–79 (2007). DOI 10.1145/1345448.1345465
10. van den Berg, R., Kipf, T.N., Welling, M.: Graph convolutional matrix completion. In: ICSCA. ACM (2018)
11. Bostandjiev, S., O'Donovan, J., Höllerer, T.: Tasteweights: a visual interactive hybrid recommender system. In: Proceedings of the ACM Conference on Recommender Systems, pp. 35–42. ACM (2012). DOI 10.1145/2365952.2365964
12. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever,

F. Liu et al., *Multimodal Learning toward Recommendation*,
https://doi.org/10.1007/978-3-031-83188-1

I., Amodei, D.: Language models are few-shot learners. In: Proceedings of the International Conference on Neural Information Processing Systems. CAI. (2020)
13. Cai, D., Qian, S., Fang, Q., Xu, C.: Heterogeneous hierarchical feature aggregation network for personalized micro-video recommendation pp. 1–1 (2021). DOI 10.1109/TMM.2021.3059508
14. Cai, D., Qian, S., Fang, Q., Xu, C.: Heterogeneous hierarchical feature aggregation network for personalized micro-video recommendation. IEEE Transactions on Multimedia **24**, 805–818 (2021). DOI 10.1109/TMM.2021.3059508
15. Catherine, R., Cohen, W.: Transnets: Learning to transform for recommendation. In: Proceedings of the ACM Conference on Recommender Systems, p. 288–296. ACM (2017). DOI 10.1145/3109859.3109878
16. Chen, C., Zhang, M., Liu, Y., Ma, S.: Neural attentional rating regression with review-level explanations. In: Proceedings of the Web Conference, pp. 1583–1592. IW3C2 (2018). DOI 10.1145/3178876.3186070
17. Chen, D., Liu, J., Dai, W., Wang, B.: Visual instruction tuning with polite flamingo. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 17745–17753. AAAI (2024). DOI 10.1609/AAAI.V38I16.29727
18. Chen, H., Li, Y., Sun, X., Xu, G., Yin, H.: Temporal meta-path guided explainable recommendation. In: Proceedings of the ACM International Conference on Web Search and Data Mining, pp. 1056–1064. ACM (2021). DOI 10.1145/3437963.3441762
19. Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., He, X.: Bias and debias in recommender system: A survey and future directions. Transaction on Information System **41**(3), 67:1–67:39 (2023). DOI 10.1145/3564284
20. Chen, J., Zhang, H., He, X., Nie, L., Liu, W., Chua, T.S.: Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 335–344. ACM (2017). DOI 10.1145/3077136.3080797
21. Chen, J., Zhuang, F., Hong, X., Ao, X., Xie, X., He, Q.: Attention-driven factor model for explainable personalized recommendation. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 909–912. ACM (2018). DOI 10.1145/3209978.3210083
22. Chen, L., Pu, P.: Critiquing-based recommenders: survey and emerging trends. User Modeling and User-Adapted Interaction **22**(1-2), 125–150 (2012). DOI 10.1007/S11257-011-9108-6
23. Chen, X., Chen, H., Xu, H., Zhang, Y., Cao, Y., Qin, Z., Zha, H.: Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 765–774. ACM (2019). DOI 10.1145/3331184.3331254
24. Cheng, H.T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G., Chai, W., Ispir, M., et al.: Wide & deep learning for recommender systems. In: Proceedings of the Workshop on Deep Learning for Recommender Systems, pp. 7–10. ACM (2016). DOI 10.1145/2988450.2988454
25. Cheng, Z., Chang, X., Zhu, L., Kanjirathinkal, R.C., Kankanhalli, M.: Mmalfm: Explainable recommendation by leveraging reviews and images. ACM Transactions on Information Systems **37**(2), 16 (2019). DOI 10.1145/3291060
26. Cheng, Z., Ding, Y., He, X., Zhu, L., Song, X., Kankanhalli, M.: A^3ncf: An adaptive aspect attention model for rating prediction. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 3748–3754. AAAI Press (2018)
27. Cheng, Z., Ding, Y., Zhu, L., Kankanhalli, M.: Aspect-aware latent factor model: Rating prediction with ratings and reviews. In: Proceedings of the Web Conference, pp. 639–648. IW3C2 (2018). DOI 10.1145/3178876.3186145
28. Cheng, Z., Han, S., Liu, F., Zhu, L., Gao, Z., Peng, Y.: Multi-behavior recommendation with cascading graph convolution networks. In: Proceedings of the Web Conference, p. 1181–1189. ACM (2023). DOI 10.1145/3543507.3583439

29. Cheng, Z., Liu, F., Mei, S., Guo, Y., Zhu, L., Nie, L.: Feature-level attentive icf for recommendation. ACM Transactions on Information Systems **40**(4), 1–24 (2022). DOI 10.1145/3490477
30. Chin, J.Y., Zhao, K., Joty, S., Cong, G.: ANR: Aspect-based neural recommender. In: Proceedings of the ACM International Conference on Information & Knowledge Management, pp. 147–156. ACM (2018). DOI 10.1145/3269206.3271810
31. Cogswell, M., Ahmed, F., Girshick, R., Zitnick, L., Batra, D.: Reducing overfitting in deep networks by decorrelating representations. arXiv e-prints (2015)
32. Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., Taylor, J.G.: Emotion recognition in human-computer interaction. IEEE Signal Processing Magazine **18**(1), 32–80 (2001). DOI 10.1109/79.911197
33. Cui, Y., Sun, H., Zhao, Y., Yin, H., Zheng, K.: Sequential-knowledge-aware next poi recommendation: A meta-learning approach. ACM Transactions on Information Systems **40**(2), 1–22 (2021). DOI 10.1145/3460198
34. Derevensky, J.L.: Modal preferences and strengths: Implications for reading research. Journal of Reading Behavior pp. 7–23 (1978)
35. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics, pp. 4171–4186. ACL (2019). DOI 10.18653/V1/N19-1423
36. Ding, Y., Ma, Y., Wong, W.K., Chua, T.: Modeling instant user intent and content-level transition for sequential fashion recommendation. IEEE Transactions on Multimedia **24**, 2687–2700 (2022). DOI 10.1109/TMM.2021.3088281
37. Du, X., Wang, X., He, X., Li, Z., Tang, J., Chua, T.S.: How to learn item representation for cold-start multimedia recommendation? In: Proceedings of the ACM International Conference on Multimedia, pp. 3469–3477 (2020). DOI 10.1145/3394171.3413628
38. Du, X., Wu, Z., Feng, F., He, X., Tang, J.: Invariant representation learning for multimedia recommendation. In: Proceedings of the ACM International Conference on Multimedia, pp. 619–628 (2022). DOI 10.1145/3503161.3548405
39. Ebesu, T., Shen, B., Fang, Y.: Collaborative memory network for recommendation systems. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 515–524. ACM (2018). DOI 10.1145/3209978.3209991
40. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the International Conference on Machine Learning, pp. 1126–1135. PMLR (2017). URL http://proceedings.mlr.press/v70/finn17a.html
41. Fu, Y., Zhang, L., Wang, J., Fu, Y., Jiang, Y.G.: Depth guided adaptive meta-fusion network for few-shot video recognition. In: Proceedings of the ACM International Conference on Multimedia, pp. 1142–1151 (2020)
42. Gao, C., Huang, C., Lin, D., Jin, D., Li, Y.: DPLCF: differentially private local collaborative filtering. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 961–970. ACM (2020). DOI 10.1145/3397271.3401053
43. Gao, C., Li, S., Zhang, Y., Chen, J., Li, B., Lei, W., Jiang, P., He, X.: Kuairand: An unbiased sequential recommendation dataset with randomly exposed videos. In: Proceedings of the ACM International Conference on Information & Knowledge Management, pp. 3953–3957. ACM (2022). DOI 10.1145/3511808.3557624
44. Gao, L., Ma, X., Lin, J., Callan, J.: Precise zero-shot dense retrieval without relevance labels. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, pp. 1762–1777. Association for Computational Linguistics (2023). DOI 10.18653/V1/2023.ACL-LONG.99
45. Gao, L., Yang, H., Wu, J., Zhou, C., Lu, W., Hu, Y.: Recommendation with multi-source heterogeneous information. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 3378–3384. AAAI Press (2018). DOI 10.24963/IJCAI.2018/469
46. Ge, Y., Liu, S., Gao, R., Xian, Y., Li, Y., Zhao, X., Pei, C., Sun, F., Ge, J., Ou, W., Zhang, Y.: Towards long-term fairness in recommendation. In: Proceedings of the ACM International

Conference on Web Search and Data Mining, pp. 445–453. ACM (2021). DOI 10.1145/3437963.3441824
47. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: Proceedings of the International Conference on Artificial Intelligence and Statistics, pp. 315–323. JMLR (2011)
48. Gönen, M., Alpaydın, E.: Multiple kernel learning algorithms. Proceedings of the International Conference on Machine Learning **12**, 2211–2268 (2011). DOI 10.5555/1953048.2021071
49. Hamilton, W.L., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 1024–1034. CAI. (2017). DOI 10.5555/3294771.3294869
50. Han, K., Xiao, A., Wu, E., Guo, J., Xu, C., Wang, Y.: Transformer in transformer. Proceedings of the International Conference on Neural Information Processing Systems **34**, 15908–15919 (2021). DOI 10.5555/3540261.3541478
51. Harper, F.M., Xu, F., Kaur, H., Condiff, K., Chang, S., Terveen, L.G.: Putting users in control of their recommendations. In: Proceedings of the ACM Conference on Recommender Systems, pp. 3–10. ACM (2015). DOI 10.1145/2792838.2800179
52. Harte, J., Zorgdrager, W., Louridas, P., Katsifodimos, A., Jannach, D., Fragkoulis, M.: Leveraging large language models for sequential recommendation. In: Proceedings of the ACM Conference on Recommender Systems, pp. 1096–1102. ACM (2023). DOI 10.1145/3604915.3610639
53. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 770–778. IEEE Computer Society (2016). DOI 10.1109/CVPR.2016.90
54. He, R., McAuley, J.: Vbpr: visual bayesian personalized ranking from implicit feedback. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 144–150. AAAI Press (2016). DOI 10.5555/3015812.3015834
55. He, X., Chen, T., Kan, M.Y., Chen, X.: Trirank: Review aware explainable recommendation by modeling aspects. In: Proceedings of the ACM International on Conference on Information and Knowledge Management, pp. 1661–1670. ACM (2015). DOI 10.1145/2806416.2806504
56. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: Lightgcn: Simplifying and powering graph convolution network for recommendation. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 639–648. ACM (2020). DOI 10.1145/3397271.3401063
57. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of the International Conference on World Wide Web, pp. 173–182. IW3C2 (2017). DOI 10.1145/3038912.3052569
58. Hershey, S., Chaudhuri, S., Ellis, D.P., Gemmeke, J.F., Jansen, A., Moore, R.C., Plakal, M., Platt, D., Saurous, R.A., Seybold, B., et al.: Cnn architectures for large-scale audio classification. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 131–135. IEEE (2017). DOI 10.1109/ICASSP.2017.7952132
59. Higgins, I., Matthey, L., Pal, A., Burgess, C.P., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework. In: Proceedings of the International Conference on Learning Representations, pp. 1–13. OpenReview.net (2017)
60. Hijikata, Y., Kai, Y., Nishida, S.: The relation between user intervention and user satisfaction for information recommendation. In: Proceedings of the ACM Symposium on Applied Computing, pp. 2002–2007. ACM (2012). DOI 10.1145/2245276.2232109
61. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. stat **1050**, 9 (2015)
62. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Computation **9**(8), 1735–1780 (1997). DOI 10.1162/neco.1997.9.8.1735
63. Hsieh, C.K., Yang, L., Cui, Y., Lin, T.Y., Belongie, S., Estrin, D.: Collaborative metric learning. In: Proceedings of the International Conference on World Wide Web, pp. 193–201. IW3C2 (2017). DOI 10.1145/3038912.3052639

64. Hsieh, J., Liu, B., Huang, D., Fei-Fei, L., Niebles, J.C.: Learning to decompose and disentangle representations for video prediction. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 515–524. CAI. (2018)
65. Hu, X., Yang, K., Fei, L., Wang, K.: Acnet: Attention based network to exploit complementary features for rgbd semantic segmentation. In: Proceedings of the International Conference on Image Processing, pp. 1440–1444. IEEE (2019). DOI 10.1109/ICIP.2019.8803025
66. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Proceedings of the International Conference on Data Mining, pp. 263–272. IEEE (2008). DOI 10.1109/ICDM.2008.22
67. Huang, P.S., He, X., Gao, J., Deng, L., Acero, A., Heck, L.: Learning deep structured semantic models for web search using click through data. In: Proceedings of the ACM International Conference on Information & Knowledge Management, pp. 2333–2338. ACM (2013). DOI 10.1145/2505515.2505665
68. Huang, Y., Zhao, F., Gui, X., Jin, H.: Path-enhanced explainable recommendation with knowledge graphs. Proceedings of the Web Conference **24**(5), 1769–1789 (2021). DOI 10.1007/S11280-021-00912-4
69. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems **20**(4), 422–446 (2002). DOI 10.1145/582415.582418
70. Ji, W., Liu, X., Zhang, A., Wei, Y., Ni, Y., Wang, X.: Online distillation-enhanced multimodal transformer for sequential recommendation. In: Proceedings of the ACM International Conference on Multimedia, pp. 955–965. ACM (2023). DOI 10.1145/3581783.3612091
71. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the ACM International Conference on Multimedia, pp. 675–678. ACM (2014). DOI 10.1145/2647868.2654889
72. Jugovac, M., Jannach, D.: Interacting with recommenders - overview and research directions. ACM Transactions on Interactive Intelligent Systems **7**(3), 10:1–10:46 (2017). DOI 10.1145/3001837
73. Kalantidis, Y., Kennedy, L., Li, L.J.: Getting the look: clothing recognition and segmentation for automatic product suggestions in everyday photos. In: Proceedings of the International Conference on Multimedia Retrieval, pp. 105–112 (2013)
74. Kang, S., Hwang, J., Kweon, W., Yu, H.: De-rrd: A knowledge distillation framework for recommender system. In: Proceedings of the ACM International Conference on Information & Knowledge Management, p. 605–614. ACM (2020). DOI 10.1145/3340531.3412005
75. Khoshneshin, M., Street, W.N.: Collaborative filtering via euclidean embedding. In: Proceedings of the ACM Conference on Recommender Systems, pp. 87–94. ACM (2010). DOI 10.1145/1864708.1864728
76. Kim, S., Kang, H., Choi, S., Kim, D., Yang, M., Park, C.: Large language models meet collaborative filtering: An efficient all-round llm-based recommender system. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1395–1406. ACM (2024). DOI 10.1145/3637528.3671931
77. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Proceedings of the International Conference on Learning Representations, pp. 1–15. OpenReview.net (2015)
78. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: Proceeding of the International Conference on Learning Representations, pp. 1–14. OpenReview.net (2014)
79. Knijnenburg, B.P., Bostandjiev, S., O'Donovan, J., Kobsa, A.: Inspectability and control in social recommenders. In: Proceedings of the ACM Conference on Recommender Systems, pp. 43–50. ACM (2012). DOI 10.1145/2365952.2365966
80. Knijnenburg, B.P., Reijmer, N.J.M., Willemsen, M.C.: Each to his own: how different users call for different interaction methods in recommender systems. In: Proceedings of the ACM Conference on Recommender Systems, pp. 141–148. ACM (2011). DOI 10.1145/2043932.2043960
81. Kolesnikov, A., Dosovitskiy, A., Weissenborn, D., Heigold, G., Uszkoreit, J., Beyer, L., Minderer, M., Dehghani, M., Houlsby, N., Gelly, S., Unterthiner, T., Zhai, X.: An image is

worth 16x16 words: Transformers for image recognition at scale. In: Proceedings of the International Conference on Learning Representations, pp. 1–21. OpenReview.net (2021)
82. Komodakis, N., Zagoruyko, S.: Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. In: Proceedings of the International Conference on Learning Representations, pp. 1–13. OpenReview.net (2017)
83. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 30–37 (2009). DOI 10.1109/MC.2009.263
84. Lan, Z.z., Bao, L., Yu, S.I., Liu, W., Hauptmann, A.G.: Multimedia classification and event detection using double fusion. Multimedia Tools and Applications **71**(1), 333–347 (2014). DOI 10.1007/s11042-013-1391-2
85. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: Proceedings of the International Conference on Machine Learning, pp. 1188–1196. JMLR (2014). DOI 10.5555/3044805.3045025
86. Lee, H., Im, J., Jang, S., Cho, H., Chung, S.: Melu: Meta-learned user preference estimator for cold-start recommendation. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1073–1082 (2019). DOI 10.1145/3292500.3330859
87. woong Lee, J., Choi, M., Lee, J., Shim, H.: Collaborative distillation for top-n recommendation. Proceedings of the International Conference on Data Mining **2019**, 369–378 (2019). DOI 10.1109/ICDM.2019.00047
88. Lewis, P.S.H., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Proceedings of the International Conference on Neural Information Processing Systems. CAI. (2020)
89. Li, A., Cheng, Z., Liu, F., Gao, Z., Guan, W., Peng, Y.: Disentangled graph neural networks for session-based recommendation. IEEE Transactions on Knowledge and Data Engineering **35**(8), 7870–7882 (2023). DOI 10.1109/TKDE.2022.3208782
90. Li, C., Quan, C., Peng, L., Qi, Y., Deng, Y., Wu, L.: A capsule network for recommendation and explaining what you like and dislike. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 275–284. ACM (2019). DOI 10.1145/3331184.3331216
91. Li, L., Zhang, Y., Chen, L.: Personalized transformer for explainable recommendation. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics, pp. 4947–4957. ACL (2021). DOI 10.18653/V1/2021.ACL-LONG.383
92. Li, L., Zhang, Y., Chen, L.: Personalized prompt learning for explainable recommendation. ACM Transactions on Information Systems **41**(4), 103:1–103:26 (2023). DOI 10.1145/3580488
93. Li, Y., Chen, H., Fu, Z., Ge, Y., Zhang, Y.: User-oriented fairness in recommendation. In: Proceedings of the Web Conference, pp. 624–632. ACM / IW3C2 (2021). DOI 10.1145/3442381.3449866
94. Li, Y., Chen, H., Li, Y., Li, L., Yu, P.S., Xu, G.: Reinforcement learning based path exploration for sequential explainable recommendation. IEEE Transactions on Knowledge and Data Engineering **35**(11), 11801–11814 (2023). DOI 10.1109/TKDE.2023.3237741
95. Li, Y., Chen, H., Xu, S., Ge, Y., Tan, J., Liu, S., Zhang, Y.: Fairness in recommendation: Foundations, methods, and applications. Transaction on Intelligent Systems Technology **14**(5), 95:1–95:48 (2023). DOI 10.1145/3610302
96. Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A., Newman, B., Yuan, B., Yan, B., Zhang, C., Cosgrove, C., Manning, C.D., Ré, C., Acosta-Navas, D., Hudson, D.A., Zelikman, E., Durmus, E., Ladhak, F., Rong, F., Ren, H., Yao, H., Wang, J., Santhanam, K., Orr, L.J., Zheng, L., Yüksekgönül, M., Suzgun, M., Kim, N., Guha, N., Chatterji, N.S., Khattab, O., Henderson, P., Huang, Q., Chi, R., Xie, S.M., Santurkar, S., Ganguli, S., Hashimoto, T., Icard, T., Zhang, T., Chaudhary, V., Wang, W., Li, X., Mai, Y., Zhang, Y., Koreeda, Y.: Holistic evaluation of language models. In: Proceedings of the International Conference on Learning Representations. OpenReview.net (2023)

97. Liao, J., Li, S., Yang, Z., Wu, J., Yuan, Y., Wang, X., He, X.: Llara: Large language-recommendation assistant. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 1785–1795. ACM (2024). DOI 10.1145/3626772.3657690
98. Lin, J., Shan, R., Zhu, C., Du, K., Chen, B., Quan, S., Tang, R., Yu, Y., Zhang, W.: Rella: Retrieval-enhanced large language models for lifelong sequential behavior comprehension in recommendation. In: Proceedings of the Web Conference, pp. 3497–3508. ACM (2024). DOI 10.1145/3589334.3645467
99. Liu, F., Chen, H., Cheng, Z., Liu, A., Nie, L., Kankanhalli, M.: Disentangled multimodal representation learning for recommendation. IEEE Transactions on Multimedia **25**, 7149–7159 (2023). DOI 10.1109/TMM.2022.3217449
100. Liu, F., Cheng, Z., Chen, H., Wei, Y., Nie, L., Kankanhalli, M.: Privacy-preserving synthetic data generation for recommendation systems. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 1379–1389. ACM (2022). DOI 10.1145/3477495.3532044
101. Liu, F., Cheng, Z., Sun, C., Wang, Y., Nie, L., Kankanhalli, M.: User diverse preference modeling by multimodal attentive metric learning. In: Proceedings of the ACM International Conference on Multimedia, p. 1526–1534. ACM (2019). DOI 10.1145/3343031.3350953
102. Liu, F., Cheng, Z., Zhu, L., Gao, Z., Nie, L.: Interest-aware message-passing gcn for recommendation. In: Proceedings of the Web Conference, p. 1296–1305. ACM (2021). DOI 10.1145/3442381.3449986
103. Liu, F., Cheng, Z., Zhu, L., Liu, C., Nie, L.: An attribute-aware attentive gcn model for attribute missing in recommendation. IEEE Transactions on Knowledge and Data Engineering **34**(9), 4077–4088 (2022). DOI 10.1109/TKDE.2020.3040772
104. Liu, F., Liu, Y., Chen, H., Cheng, Z., Nie, L., Kankanhalli, M.: Understanding before recommendation: Semantic aspect-aware review exploitation via large language models. ACM Transactions on Information Systems pp. 1–26 (2024). DOI 10.1145/3704999
105. Liu, F., Zhao, S., Cheng, Z., Nie, L., Kankanhalli, M.: Cluster-based graph collaborative filtering. ACM Transactions on Information Systems **42**(6), 1–24 (2024). DOI 10.1145/3687481
106. Liu, H., He, X., Feng, F., Nie, L., Liu, R., Zhang, H.: Discrete factorization machines for fast feature-based recommendation. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 3449–3455. AAAI Press (2018). DOI 10.24963/IJCAI.2018/479
107. Liu, H., Li, C., Li, Y., Lee, Y.J.: Improved baselines with visual instruction tuning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 26286–26296. IEEE Computer Society (2024). DOI 10.1109/CVPR52733.2024.02484
108. Liu, H., Wei, Y., Yin, J., Nie, L.: Hs-gcn: Hamming spatial graph convolutional networks for recommendation. IEEE Transactions on Knowledge and Data Engineering **35**(6), 5977–5990 (2023). DOI 10.1109/TKDE.2022.3158317
109. Liu, M., Wang, X., Nie, L., He, X., Chen, B., Chua, T.S.: Attentive moment retrieval in videos. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 15–24. ACM (2018). DOI 10.1145/3209978.3210003
110. Liu, M., Wang, X., Nie, L., Tian, Q., Chen, B., Chua, T.S.: Cross-modal moment localization in videos. In: Proceedings of the ACM International Conference on Multimedia, pp. 843–851. ACM (2018). DOI 10.1145/3240508.3240549
111. Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., Neubig, G.: Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. ACM Computing Surveys **55**(9), 195:1–195:35 (2023). DOI 10.1145/3560815
112. Liu, Q., Hu, J., Xiao, Y., Zhao, X., Gao, J., Wang, W., Li, Q., Tang, J.: Multimodal recommender systems: A survey. ACM Computing Survey **57**(2) (2024). DOI 10.1145/3695461
113. Liu, S., Chen, Z., Liu, H., Hu, X.: User-video co-attention network for personalized micro-video recommendation. In: Proceedings of the Web Conference, pp. 3020–3026. ACM (2019). DOI 10.1145/3308558.3313513

114. Liu, X., Tao, Z., Shao, J., Yang, L., Huang, X.: Elimrec: Eliminating single-modal bias in multimedia recommendation. In: Proceedings of the ACM International Conference on Multimedia, pp. 687–695. ACM (2022). DOI 10.1145/3503161.3548404
115. Liu, Y., Wang, X., Wu, S., Xiao, Z.: Independence promoted graph disentangled networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp. 4916–4923. AAAI (2020). DOI 10.1609/AAAI.V34I04.5929
116. Liu, Y., Wang, Y., Sun, L., Yu, P.S.: Rec-gpt4v: Multimodal recommendation with large vision-language models. CoRR **abs/2402.08670** (2024). DOI 10.48550/ARXIV.2402.08670
117. Ma, J., Cui, P., Kuang, K., Wang, X., Zhu, W.: Disentangled graph convolutional networks. In: Proceedings of the International Conference on Machine Learning, pp. 4212–4221. PMLR (2019)
118. Ma, J., Zhou, C., Cui, P., Yang, H., Zhu, W.: Learning disentangled representations for recommendation. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 5712–5723. CAI. (2019)
119. Ma, W., Zhang, M., Cao, Y., Jin, W., Wang, C., Liu, Y., Ma, S., Ren, X.: Jointly learning explainable rules for recommendation with knowledge graph. In: Proceedings of the Web Conference, pp. 1210–1221. ACM (2019). DOI 10.1145/3308558.3313607
120. van der Maaten, L., Hinton, G.: Visualizing data using t-SNE. Journal of Machine Learning Research **9**, 2579–2605 (2008)
121. McAuley, J., Leskovec, J.: Hidden factors and hidden topics: understanding rating dimensions with review text. In: Proceedings of the ACM Conference on Recommender Systems, pp. 165–172. ACM (2013). DOI 10.1145/2507157.2507163
122. McAuley, J., Targett, C., Shi, Q., Van Den Hengel, A.: Image-based recommendations on styles and substitutes. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 43–52. ACM (2015). DOI 10.1145/2766462.2767755
123. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of the International Conference on Learning Representations. OpenReview.net (2013)
124. Min, W., Jiang, S., Jain, R.: Food recommendation: Framework, existing solutions, and challenges. In: IEEE Transactions on Multimedia, pp. 2659–2671 (2020). DOI 10.1109/TMM.2019.2958761
125. Mroueh, Y., Marcheret, E., Goel, V.: Deep multimodal learning for audio-visual speech recognition. In: Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2130–2134. IEEE (2015). DOI 10.1109/ICASSP.2015.7178347
126. Mu, S., Li, Y., Zhao, W.X., Li, S., Wen, J.: Knowledge-guided disentangled representation learning for recommender systems. ACM Transactions on Information Systems **40**(1), 6:1–6:26 (2022). DOI 10.1145/3464304
127. Mukherjee, P., Das, A., Bhunia, A.K., Roy, P.P.: Cogni-net: Cognitive feature learning through deep visual perception. In: Proceedings of the International Conference on Image Processing, pp. 4539–4543. IEEE (2019). DOI 10.1109/ICIP.2019.8803717
128. Munkhdalai, T., Yu, H.: Meta networks. In: Proceedings of the International Conference on Machine Learning, pp. 2554–2563. JMLR (2017). DOI 10.5555/3305890.3305945
129. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: Proceedings of the International Conference on Machine Learning, pp. 689–696 (2011)
130. Nie, L., Song, X., Chua, T.S.: Learning from multiple social networks. Synthesis Lectures on Information Concepts Retrieval & Services pp. 1–118 (2016). DOI 10.2200/S00714ED1V01Y201603ICR048
131. Nie, L., Wang, W., Hong, R., Wang, M., Tian, Q.: Multimodal dialog system: Generating responses via adaptive decoders. In: Proceedings of the ACM International Conference on Multimedia, pp. 1098–1106. ACM (2019). DOI 10.1145/3343031.3350923
132. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: Pytorch: An imperative style,

high-performance deep learning library. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 8024–8035. CAI. (2019)
133. Ram, O., Levine, Y., Dalmedigos, I., Muhlgay, D., Shashua, A., Leyton-Brown, K., Shoham, Y.: In-context retrieval-augmented language models. Transactions of the Association for Computational Linguistics **11**, 1316–1331 (2023). DOI 10.1162/TACL_A_00605
134. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence, pp. 452–461. AUAI Press (2009)
135. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. In: Proceedings of the International Conference on Learning Representations, pp. 1–13. OpenReview.net (2015)
136. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: Proceedings of the International Conference on Neural Information Processing Systems, p. 1257–1264. CAI. (2007)
137. Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., Lillicrap, T.: Meta-learning with memory-augmented neural networks. In: Proceedings of the International Conference on Machine Learning, pp. 1842–1850. PMLR (2016)
138. Schafer, J.B., Konstan, J.A., Riedl, J.: Meta-recommendation systems: user-controlled integration of diverse recommendations. In: Proceedings of the ACM International Conference on Information & Knowledge Management, pp. 43–51. ACM (2002). DOI 10.1145/584792.584803
139. Schaffer, J., Höllerer, T., O'Donovan, J.: Hypothetical recommendation: A study of interactive profile manipulation behavior for recommender systems. In: Proceedings of the International Florida Artificial Intelligence Research Society Conference, pp. 507–512. AAAI Press (2015)
140. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.M.: Methods and metrics for cold-start recommendations. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 253–260. ACM (2002). DOI 10.1145/564376.564421
141. Seo, S., Huang, J., Yang, H., Liu, Y.: Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In: Proceedings of the ACM Conference on Recommender Systems, pp. 297–305. ACM (2017). DOI 10.1145/3109859.3109890
142. Shang, Y., Gao, C., Chen, J., Jin, D., Li, Y.: Improving item-side fairness of multimodal recommendation via modality debiasing. In: Proceedings of the Web Conference, pp. 4697–4705. ACM (2024). DOI 10.1145/3589334.3648156
143. Shao, Z., Gong, Y., Shen, Y., Huang, M., Duan, N., Chen, W.: Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In: Findings of the Association for Computational Linguistics, pp. 9248–9274. ACL (2023). DOI 10.18653/V1/2023.FINDINGS-EMNLP.620
144. Sharma, A., Cosley, D.: Do social explanations work?: studying and modeling the effects of social explanations in recommender systems. In: Proceedings of the Web Conference, pp. 1133–1144. ACM (2013). DOI 10.1145/2488388.2488487
145. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Proceedings of the International Conference on Learning Representations, pp. 1–14. OpenReview.net (2015)
146. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 4080–4090. CAI. (2017)
147. Song, X., Feng, F., Han, X., Yang, X., Liu, W., Nie, L.: Neural compatibility modeling with attentive knowledge distillation. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 5–14. ACM (2018). DOI 10.1145/3209978.3209996
148. Srivastava, N., Salakhutdinov, R.R.: Multimodal learning with deep boltzmann machines. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 2222–2230. CAI. (2012)
149. Sun, Y., Zhang, Y.: Conversational recommender system. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 235–244. ACM (2018). DOI 10.1145/3209978.3210002

150. Swearingen, K., Sinha, R.: Beyond algorithms: An hci perspective on recommender systems. In: ACM SIGIR 2001 workshop on recommender systems, vol. 13, pp. 1–11 (2001)
151. Székely, G.J., Rizzo, M.L.: Brownian distance covariance. In: The Annals of Applied Statistics, pp. 1236–1265. IMS (2009)
152. Székely, G.J., Rizzo, M.L., Bakirov, N.K.: Measuring and testing dependence by correlation of distances. In: The Annals of Statistics, pp. 2769–2794. IMS (2007)
153. Tan, J., Xu, S., Ge, Y., Li, Y., Chen, X., Zhang, Y.: Counterfactual explainable recommendation. In: Proceedings of the ACM International Conference on Information & Knowledge Management, pp. 1784–1793. ACM (2021). DOI 10.1145/3459637.3482420
154. Tan, J., Xu, S., Hua, W., Ge, Y., Li, Z., Zhang, Y.: Idgenrec: Llm-recsys alignment with textual ID learning. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 355–364. ACM (2024). DOI 10.1145/3626772.3657821
155. Tan, Y., Zhang, M., Liu, Y., Ma, S.: Rating-boosted latent topics: Understanding users and items with ratings and reviews. In: Proceedings of the International Joint Conference on Artificial Intelligence, p. 2640–2646. AAAI Press (2016)
156. Tang, J., Wang, K.: Ranking distillation: Learning compact ranking models with high performance for recommender system. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 2289–2298 (2018). DOI 10.1145/3219819.3220021
157. Tay, Y., Anh Tuan, L., Hui, S.C.: Latent relational metric learning via memory-based attention for collaborative ranking. In: Proceedings of the Web Conference, pp. 729–739. IW3C2 (2018). DOI 10.1145/3178876.3186154
158. Tian, Y., Krishnan, D., Isola, P.: Contrastive representation distillation. In: Proceedings of the International Conference on Learning Representations, pp. 1–19. OpenReview.net (2020)
159. Tran, N.T., Lauw, H.W.: Aligning dual disentangled user representations from ratings and textual content. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, p. 1798–1806. ACM (2022). DOI 10.1145/3534678.3539474
160. Tversky, A., Gati, I.: Similarity, separability, and the triangle inequality. Psychological review **89**(2), 123 (1982)
161. Vartak, M., Thiagarajan, A., Miranda, C., Bratman, J., Larochelle, H.: A meta-learning perspective on cold-start recommendations for items. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 6907–6917. CAI. (2017). DOI 10.5555/3295222.3295434
162. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: Proceedings of the International Conference on Learning Representations, pp. 1–12. OpenReview.net (2019)
163. Wang, H., Lai, C., Sun, Y., Ge, W.: Weakly supervised gaussian contrastive grounding with large multimodal models for video question answering. In: Proceedings of the ACM International Conference on Multimedia, pp. 5289–5298. ACM (2024). DOI 10.1145/3664647.3680826
164. Wang, J., Tang, Q.: Differentially private neighborhood-based recommender systems. In: International Conference on ICT Systems Security and Privacy Protection, *IFIP Advances in Information and Communication Technology*, vol. 502, pp. 459–473. Springer (2017). DOI 10.1007/978-3-319-58469-0_31
165. Wang, M., Gao, Y., Lu, K., Rui, Y.: View-based discriminative probabilistic modeling for 3d object retrieval and recognition. ACM Transactions on Information Systems **22**, 1395–1407 (2013). DOI 10.1109/TIP.2012.2207397
166. Wang, M., Luo, C., Ni, B., Yuan, J., Wang, J., Yan, S.: First-person daily activity recognition with manipulated object proposals and non-linear feature fusion. IEEE Transactions on Circuits and Systems for Video Technology **28**, 2946–2955 (2018). DOI 10.1109/TCSVT.2017.2716819
167. Wang, N., Wang, H., Jia, Y., Yin, Y.: Explainable recommendation via multi-task learning in opinionated text data. In: Proceedings of the International ACM SIGIR Conference on

Research and Development in Information Retrieval, pp. 165–174. ACM (2018). DOI 10.1145/3209978.3210010
168. Wang, S., Wang, Y., Tang, J., Shu, K., Ranganath, S., Liu, H.: What your images reveal: Exploiting visual contents for point-of-interest recommendation. In: Proceedings of the Web Conference, pp. 391–400. IW3C2 (2017). DOI 10.1145/3038912.3052638
169. Wang, W., Feng, F., He, X., Zhang, H., Chua, T.S.: Clicks can be cheating: Counterfactual recommendation for mitigating clickbait issue. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1288–1297 (2021). DOI 10.1145/3404835.3462962
170. Wang, W., Feng, F., Nie, L., Chua, T.S.: User-controllable recommendation against filter bubbles. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1251–1261. ACM (2022). DOI 10.1145/3477495.3532075
171. Wang, W., Lin, X., Feng, F., He, X., Chua, T.S.: Generative recommendation: Towards next-generation recommender paradigm. arXiv preprint arXiv:2304.03516 (2023)
172. Wang, W., Lin, X., Feng, F., He, X., Lin, M., Chua, T.S.: Causal representation learning for out-of-distribution recommendation. In: Proceedings of the Web Conference, pp. 3562–3571. ACM (2022). DOI 10.1145/3485447.3512251
173. Wang, X., He, X., Wang, M., Feng, F., Chua, T.S.: Neural graph collaborative filtering. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 165–174. ACM (2019). DOI 10.1145/3331184.3331267
174. Wang, X., Jin, H., Zhang, A., He, X., Xu, T., Chua, T.S.: Disentangled graph collaborative filtering. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 1001–1010. ACM (2020). DOI 10.1145/3397271.3401137
175. Wang, Y., Ma, W., Zhang, M., Liu, Y., Ma, S.: A survey on the fairness of recommender systems. Transaction on Information System **41**(3), 52:1–52:43 (2023). DOI 10.1145/3547333
176. Wang, Y., Tang, S., Lei, Y., Song, W., Wang, S., Zhang, M.: Disenhan: Disentangled heterogeneous graph attention network for recommendation. In: Proceedings of the ACM International Conference on Information & Knowledge Management, p. 1605–1614. ACM (2020). DOI 10.1145/3340531.3411996
177. Wang, Y., Xu, X., Yu, W., Xu, R., Cao, Z., Shen, H.T.: Combine early and late fusion together: A hybrid fusion framework for image-text matching. In: Proceedings of the IEEE International Conference on Multimedia and Expo, pp. 1–6. IEEE (2021). DOI 10.1109/ICME51207.2021.9428201
178. Wasinger, R., Wallbank, J., Pizzato, L.A.S., Kay, J., Kummerfeld, B., Böhmer, M., Krüger, A.: Scrutable user models and personalised item recommendation in mobile lifestyle applications. In: User Modeling, Adaptation, and Personalization - 21th International Conference, *Lecture Notes in Computer Science*, vol. 7899, pp. 77–88. Springer (2013). DOI 10.1007/978-3-642-38844-6_7
179. Wei, J., Bosma, M., Zhao, V.Y., Guu, K., Yu, A.W., Lester, B., Du, N., Dai, A.M., Le, Q.V.: Finetuned language models are zero-shot learners. In: Proceedings of the International Conference on Learning Representations, pp. 1–46. OpenReview.net (2022)
180. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E.H., Le, Q.V., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models. In: Proceedings of the International Conference on Neural Information Processing Systems. CAI. (2022)
181. Wei, W., Ren, X., Tang, J., Wang, Q., Su, L., Cheng, S., Wang, J., Yin, D., Huang, C.: Llmrec: Large language models with graph augmentation for recommendation. In: Proceedings of the Web Conference, pp. 806–815. ACM (2024). DOI 10.1145/3616855.3635853
182. Wei, Y., Cheng, Z., Yu, X., Zhao, Z., Zhu, L., Nie, L.: Personalized hashtag recommendation for micro-videos. In: Proceedings of the ACM International Conference on Multimedia, pp. 1446–1454. ACM (2019). DOI 10.1145/3343031.3350858
183. Wei, Y., Liu, W., Liu, F., Wang, X., Nie, L., Chua, T.S.: Lightgt: A light graph transformer for multimedia recommendation. In: Proceedings of the International ACM SIGIR Conference

on Research and Development in Information Retrieval, p. 1508–1517. ACM (2023). DOI 10.1145/3539618.3591716
184. Wei, Y., Wang, X., Guan, W., Nie, L., Lin, Z., Chen, B.: Neural multimodal cooperative learning toward micro-video understanding. IEEE Transactions on Image Processing **29**, 1–14 (2019). DOI 10.1109/TIP.2019.2923608
185. Wei, Y., Wang, X., Nie, L., He, X., Chua, T.S.: Graph-refined convolutional network for multimedia recommendation with implicit feedback. In: Proceedings of the ACM International Conference on Multimedia, pp. 3541–3549. ACM (2020). DOI 10.1145/3394171.3413556
186. Wei, Y., Wang, X., Nie, L., He, X., Hong, R., Chua, T.: MMGCN: multi-modal graph convolution network for personalized recommendation of micro-video. In: Proceedings of the ACM International Conference on Multimedia, pp. 1437–1445. ACM (2019). DOI 10.1145/3343031.3351034
187. Weston, J., Bengio, S., Usunier, N.: Large scale image annotation: learning to rank with joint word-image embeddings. Machine learning **81**(1), 21–35 (2010). DOI 10.1007/S10994-010-5198-3
188. Wu, L., Quan, C., Li, C., Wang, Q., Zheng, B., Luo, X.: A context-aware user-item representation learning for item recommendation. ACM Transactions on Information Systems **37**(2), 22:1–22:29 (2019). DOI 10.1145/3298988
189. Wu, Q., Zhang, H., Gao, X., Yan, J., Zha, H.: Towards open-world recommendation: An inductive model-based collaborative filtering approach. In: Proceedings of the International Conference on Machine Learning, *Proceedings of Machine Learning Research*, vol. 139, pp. 11329–11339. PMLR (2021)
190. Xavier, G., Yoshua, B.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the International Conference on Artificial Intelligence and Statistics, pp. 249–256. PMLR (2010)
191. Xi, Y., Liu, W., Lin, J., Cai, X., Zhu, H., Zhu, J., Chen, B., Tang, R., Zhang, W., Yu, Y.: Towards open-world recommendation with knowledge augmentation from large language models. In: Proceedings of the ACM Conference on Recommender Systems, pp. 12–22. ACM (2024). DOI 10.1145/3640457.3688104
192. Xu, W., Wu, Q., Wang, R., Ha, M., Ma, Q., Chen, L., Han, B., Yan, J.: Rethinking cross-domain sequential recommendation under open-world assumptions. In: Proceedings of the Web Conference, pp. 3173–3184. ACM (2024). DOI 10.1145/3589334.3645351
193. Xue, H., Dai, X., Zhang, J., Huang, S., Chen, J.: Deep matrix factorization models for recommender systems. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 3203–3209. AAAI Press (2017). DOI 10.5555/3172077.3172336
194. Yan, M., Cheng, Z., Gao, C., Sun, J., Liu, F., Sun, F., Li, H.: Cascading residual graph convolutional network for multi-behavior recommendation. ACM Transactions on Information Systems **42**(1), 1–26 (2023). DOI 10.1145/3587693
195. Yan, M., Liu, F., Sun, J., Sun, F., Cheng, Z., Han, Y.: Behavior-contextualized item preference modeling for multi-behavior recommendation. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 946–955. ACM (2024). DOI 10.1145/3626772.3657696
196. Yang, Q.: Federated recommendation systems. In: International Conference on Big Data, p. 1. IEEE (2019). DOI 10.1109/BIGDATA47090.2019.9005952
197. Yang, S., Ma, W., Sun, P., Ai, Q., Liu, Y., Cai, M., Zhang, M.: Sequential recommendation with latent relations based on large language model. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 335–344. ACM (2024). DOI 10.1145/3626772.3657762
198. Yang, W., Fang, Z., Zhang, T., Wu, S., Lu, C.: Modal-aware bias constrained contrastive learning for multimodal recommendation. In: Proceedings of the ACM International Conference on Multimedia, pp. 6369–6378. ACM (2023). DOI 10.1145/3581783.3612568
199. Yang, W., Fang, Z., Zhang, T., Wu, S., Lu, C.: Modal-aware bias constrained contrastive learning for multimodal recommendation. In: Proceedings of the ACM International Conference on Multimedia, pp. 6369–6378. ACM (2023). DOI 10.1145/3581783.3612568

200. Yang, X., Wang, M., Tao, D.: Person re-identification with metric learning using privileged information. ACM Transactions on Information Processing **27**, 791–805 (2018). DOI 10.1109/TIP.2017.2765836
201. Ye, Y., Zheng, Z., Shen, Y., Wang, T., Zhang, H., Zhu, P., Yu, R., Zhang, K., Xiong, H.: Harnessing multimodal large language models for multimodal sequential recommendation. CoRR **abs/2408.09698** (2024). DOI 10.48550/ARXIV.2408.09698
202. Yoshua, B., Yann, L.: Scaling learning algorithms towards ai. Large-scale kernel machines **34**(5), 1–41 (2007)
203. Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z., Tay, F.E.H., Feng, J., Yan, S.: Tokens-to-token vit: Training vision transformers from scratch on imagenet. In: Proceedings of the International Conference on Computer Vision, pp. 538–547. IEEE (2021). DOI 10.1109/ICCV48922.2021.00060
204. Zhang, F., Yuan, N.J., Lian, D., Xie, X., Ma, W.Y.: Collaborative knowledge base embedding for recommender systems. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 353–362. ACM (2016). DOI 10.1145/2939672.2939673
205. Zhang, H., Yang, Y., Luan, H., Yang, S., Chua, T.S.: Start from scratch: Towards automatically identifying, modeling, and naming visual attributes. In: Proceedings of the ACM International Conference on Multimedia, pp. 187–196. ACM (2014). DOI 10.1145/2647868.2654915
206. Zhang, J., Zhu, Y., Liu, Q., Wu, S., Wang, S., Wang, L.: Mining latent structures for multimedia recommendation. In: Proceedings of the ACM International Conference on Multimedia, p. 3872–3880. ACM (2021). DOI 10.1145/3474085.3475259
207. Zhang, K., Cao, Q., Wu, Y., Sun, F., Shen, H., Cheng, X.: Lorec: Combating poisons with large language model for robust sequential recommendation. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1733–1742. ACM (2024). DOI 10.1145/3626772.3657684
208. Zhang, L., Shi, Y., Shi, Z., Ma, K., Bao, C.: Task-oriented feature distillation. In: Proceedings of the International Conference on Neural Information Processing Systems, pp. 14759–14771. CAI. (2020)
209. Zhang, S., Yao, L., Tay, Y., Xu, X., Zhang, X., Zhu, L.: Metric factorization: Recommendation beyond matrix factorization. In: ICDE. IEEE (2019)
210. Zhang, X., Xu, B., Yang, L., Li, C., Ma, F., Liu, H., Lin, H.: Price does matter! modeling price and interest preferences in session-based recommendation. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 1684–1693. ACM (2022). DOI 10.1145/3477495.3532043
211. Zhang, Y., Ai, Q., Chen, X., Croft, W.B.: Joint representation learning for top-n recommendation with heterogeneous information sources. In: Proceedings of the ACM International Conference on Information & Knowledge Management, pp. 1449–1458. ACM (2017). DOI 10.1145/3132847.3132892
212. Zhang, Y., Chen, X.: Explainable recommendation: A survey and new perspectives. Foundations and Trends in Information Retrieval **14**(1), 1–101 (2020). DOI 10.1561/1500000066
213. Zhang, Y., Cheng, Z., Liu, F., Yang, X., Peng, Y.: Decoupled domain-specific and domain-conditional representation learning for cross-domain recommendation. Information Processing & Management **61**(3), 103689 (2024). DOI 10.1016/j.ipm.2024.103689
214. Zhao, T., McAuley, J., King, I.: Improving latent factor models via personalized feature projection for one class recommendation. In: Proceedings of the ACM International Conference on Information & Knowledge Management, pp. 821–830. ACM (2015). DOI 10.1145/2806416.2806511
215. Zheng, L., Noroozi, V., Yu, P.S.: Joint deep modeling of users and items using reviews for recommendation. In: Proceedings of the ACM International Conference on Web Search and Data Mining, pp. 425–434. ACM (2017). DOI 10.1145/3018661.3018665
216. Zheng, Z., Chao, W., Qiu, Z., Zhu, H., Xiong, H.: Harnessing large language models for text-rich sequential recommendation. In: Proceedings of the Web Conference, pp. 3207–3216. ACM (2024). DOI 10.1145/3589334.3645358

217. Zhou, G., Zhu, X., Song, C., Fan, Y., Zhu, H., Ma, X., Yan, Y., Jin, J., Li, H., Gai, K.: Deep interest network for click-through rate prediction. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1059–1068. ACM (2018). DOI 10.1145/3219819.3219823
218. Zhou, X., Zhou, H., Liu, Y., Zeng, Z., Miao, C., Wang, P., You, Y., Jiang, F.: Bootstrap latent representations for multi-modal recommendation. In: Proceedings of the Web Conference, p. 845–854. ACM (2023). DOI 10.1145/3543507.3583251
219. Zhou, Y., Feng, T., Liu, M., Zhu, Z.: A generalized propensity learning framework for unbiased post-click conversion rate estimation. In: Proceedings of the ACM International Conference on Information & Knowledge Management, pp. 3554–3563. ACM (2023). DOI 10.1145/3583780.3614760
220. Zhu, M., Han, K., Zhang, C., Lin, J., Wang, Y.: Low-resolution visual recognition via deep feature distillation. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing pp. 3762–3766 (2019). DOI 10.1109/ICASSP.2019.8682926
221. Zhu, Y., Xie, R., Zhuang, F., Ge, K., Sun, Y., Zhang, X., Lin, L., Cao, J.: Learning to warm up cold item embeddings for cold-start recommendation with meta scaling and shifting networks. In: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1167–1176. ACM (2021). DOI 10.1145/3404835.3462843