User Plays a Role: User-insight Multi-modal Recommendation

Jingyu Xu, Zechao Hu, Hao Li, Caoyuan Ma, Zhengwei Yang, *Student Member, IEEE*, Zheng Wang, *Senior Member, IEEE*

***Abstract*—Multi-Modal Recommendation (MMRec) aims to help users explore their potential interested items based on multi-modal information input and has been widely used in e-commerce platforms. Recent works mainly focus on modeling item-side information. However, they ignore the abundant semantic information from the user-information modeling, including age, gender, feedback, etc. Such imbalanced attention to item and user leads to inadequate expressiveness of comprehensive interests. In this paper, we propose a novel User-insight Multi-modal recommendation framework, termed UiM. This framework improves user modeling in three aspects: Firstly, to better explore the primary interests from a large-scale item pool, we propose to construct an enriched user profile to re-distribute attention to users' historical interactions. Secondly, to further disentangle compact representations from heterogeneous items, we propose to apply multi-interest feature extraction on re-attentioned item features. Moreover, an intrinsic shortage of a trivial recommender system is that it fails to access user feedback for in-place result adjustment. As a solution, we access pseudo feedback beforehand from an intelligent agent, then accordingly perform potential adjustments to recommendation candidates for finer results. Extensive experiments show that our model outperforms state-of-the-art multi-modal recommendation models in three public datasets.**

***Index Terms*—Multi-modal Recommendation, User Profile Modeling, Simulated User**

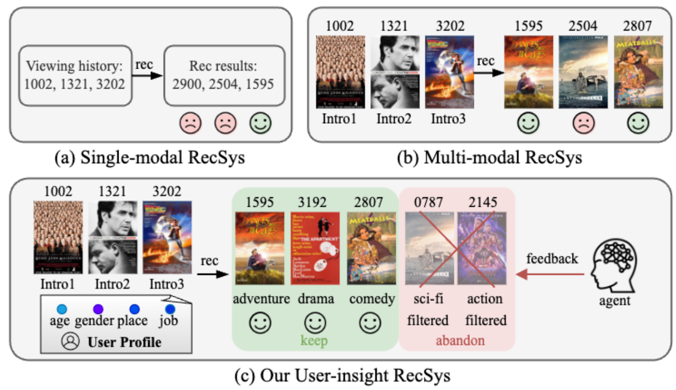

Fig. 1. The pipelines of three kinds of recommender systems. (a) single-modal recommender system, where usually only the ID embedding of historical items is fed into the system for recommendations generation; (b) multi-modal recommender system, which additionally takes multi-modal information of items as input to enhance item representation; (c) our user-insight multi-modal recommender systems, where the user profile is integrated with historical records to collectively extract the user's interests, and the system gives recommendation results after obtaining feedback from a simulated user agent with initial categorized results.

I. Introduction

PERSONALIZED recommendation [1]–[4] aims to help users explore their potential interested items and has been applied to many fields such as news recommendation, book recommendation, multi-modal recommendation, etc. A long-standing challenge in this task is dealing with data sparsity, specifically in modeling user interests from very limited historical interactions.

Many deployed recommender systems [5]–[10] utilize single-modality information to exploit item representation and implicitly mine user interests. Some models [5], [6] only take historical records as input and cannot proceed with other side information such as knowledge graphs about item-item relations, thus lacking generalization capabilities. On top of this, many methods [11]–[16] have been proposed to utilize side information to improve item representation learning. For example, KGAT [11] utilizes a Knowledge Graph (KG) to enrich the user-item bipartite graph with real-world knowledge. HPM [17] incorporates category-level information with ID attributes for better item representation learning. In some other research [13]–[16], user reviews about items are incorporated to enhance item representation learning. However, despite these improvements, single-modal methods fail to take advantage of visual features, which contain abundant latent information like fine-grained item style.

To incorporate visual features in recommender systems, numerous approaches [3], [18]–[24] have been proposed and widely applied since the rise of Multi-Modal Recommendation (MMRec) task. DeepStyle [18] incorporates style information of clothing to complement visual features in e-commerce recommendation. VECF [19] uses pre-segmented images to learn the human attention on different regions of an image. LATTICE [22] learns item-item relations to explicitly inject high-order item affinities into item representations. As a consequence, these methods perform better than single-modal ones when dealing with multi-modal inputs.

However, all of the above methods still put their primary attention on the item side, while ignoring rich latent information from the user side. Since users play a central role in personalized recommendation services, user modeling [25] is equally important to item representation learning. We

Jingyu Xu, Zechao Hu, Hao Li, Caoyuan Ma, Zhengwei Yang, and Zheng Wang are with the National Engineering Research Center for Multimedia Software, Institute of Artificial Intelligence, School of Computer Science, Wuhan University, Wuhan, 430072, China, and also with the Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan, 430072, China (e-mail: wangzwhu@whu.edu.cn). (*Corresponding author: Zheng Wang.*)

illustrate the problem in Fig. 1, where different pipelines of recommender systems are shown. In subfigure (a), for the single-modal recommender systems, only the ID embeddings of historical items (1002, 1321, 3202) are used as input, and the poor expressiveness of pure ID information leads to unsatisfied performance, with only one out of three marked as good. In subfigure (b), for current multi-modal methods, multi-modal features like item pictures and textual descriptions are integrated with ID embedding to enhance item representations. But in a real scenario, the user interests are multifold. For example, in an e-commerce dataset, a user might be interested in pet supplies and fashionable clothing simultaneously. However, the above two pipelines usually generate an overall embedding for all interacted items, thus failing to learn disentangled item representation. Moreover, due to the inherent limitation of offline training and online testing, user feedback was not accessible for previous works until the final recommendations were produced. This means that feedback can only indirectly improve the recommendation system and is not frequently updated. Consequently, they can not deal with frequent changes in user interests when inference. We summarize these drawbacks of current single- and multi-modal methods as:

- They partially focus on item-side information while ignoring the user-side information, including demographic information and dynamic behavioral labels like acceptance for new items;
- They struggle to learn effective representation for heterogeneous interacted items, where user interests and item features are entangled;
- The lack of online feedback makes current methods vulnerable to newly emerged interests, which is usual in multi-modal recommendations.

In this paper, we propose a user-insight multi-modal framework, which is enabled with comprehensive user information modeling for recommendation. Specifically, our solution contains the following three modules:

Input with Enriched User Profile. Most existing works tend to only utilize static information from user profiles, such as gender, age, OS platform, etc. We argue that these user profile features alone can not comprehensively reflect the user interests. To manufacture comprehensive user interest representation, we propose to enrich the user profile with more unexploited hidden information. Specifically, we leverage data mining to explore attributes of economic status and acceptance of new items based on explicit demographic annotations.

Multi-modality-interest Extraction. To extract users' complex interests and enhance the expressiveness of the model in encoding user interests, we have tailored the Capsule Network (CapsNet) to the field of MMRec. Originally, CapsNet was limited to extracting multiple interests within single-modality scenarios [1], [2], [7]. However, by applying CapsNet on the multi-modal item representation, we have enabled it to extract multi-modality interests, effectively adapting it for the MMRec task.

Simulated User with Real-time Feedback. Feedback is crucial to a recommender system. As a solution, we are the first to propose to introduce a virtual user simulated by a large language model (LLM) into the loop of recommendation. The virtual user functions as the spokesperson of the real user by incorporating the enriched user profile and interaction histories. In practice, the virtual user takes the recommendation candidate results as input and outputs its feedback. Finally, the recommender system can timely revise the results accordingly based on feedback.

Compared with directly using LLMs such as ChatGPT as RecSys [26], [27], the initial recommendations can utilize collaborative signals, which are of great importance in recommendation scenes and an LLM is not capable of processing such signals.

Our contributions can be summarized threefold:

- In this work, we pioneeringly explore the usage of more comprehensive user modeling in the field of MMRec. The incorporation of comprehensive user profiles in MMRec, which includes not only just demographic information but also hidden information and behavioral labels, does have a positive influence, leading to an active impact on candidate item retrieval tasks.
- From the aspect of specific techniques, we propose a novel **U**ser-**i**nsight **M**ulti-modal recommendation framework (UiM), in which we enhance user modeling in the multi-modal recommendation from three perspectives: 1) enriched user profile integration; 2) multi-modality-interest learning; and 3) pre-trained simulated user.
- We conducted extensive experiments on three real-world datasets, indicating that the proposed UiM achieves state-of-the-art performance.

II. Related Work

A. Multi-modal Recommendation

The multi-modal recommender systems [3], [4], [20]–[22], [28] take a large amount of multimedia content of items into consideration, which has been successfully applied to many scenarios in the recommender fields. Most methods enhance item representation learning with multi-modal features. For example, He et al. [28] believe that people often pay close attention to the visual information of products when shopping, so they extend MF by extracting visual features from images to improve recommendation performance. ACF [20] is based on an attention mechanism in the item and uses component layers to handle recommendation tasks in the multi-modal field. In addition, most researchers focus on designing frameworks for extracting better multi-modal features. Xu et al. [29] proposed a multi-modal recommendation model that aims to capture high-level conceptual information and explore the connection between textual and visual features between users and items. MMGCL [3] uses a self-supervised manner to disentangle the users' tastes on different modalities. Recently, Yuan et al. [12] revisits the comparison between ID-based and multi-modal recommender systems and reveals the advantages of multi-modal recommender systems in generalization capabilities. However, these multi-modal models partially focus on item-side information and fail to pay attention to the user-side information.

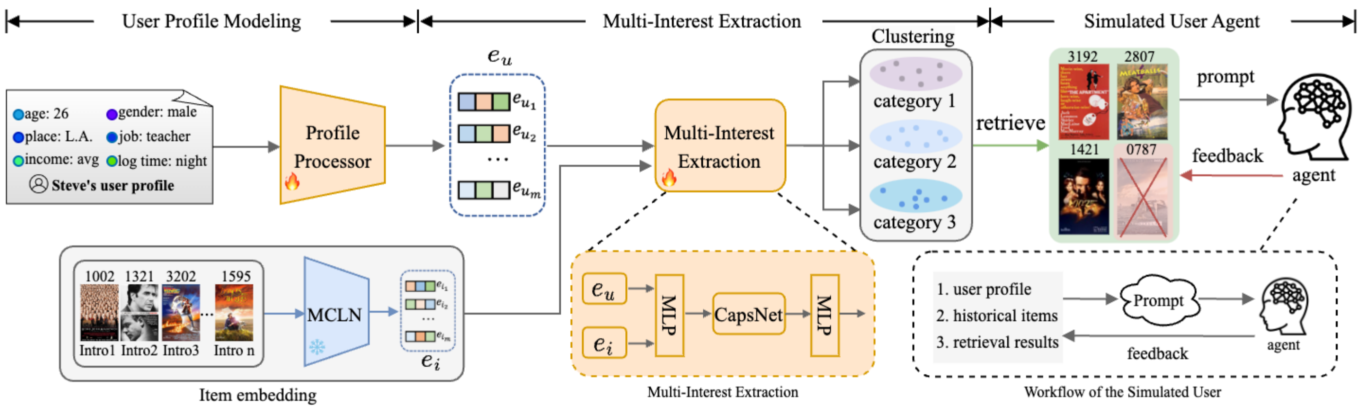

Fig. 2. Overview of our proposed framework. Different from previous works that only use historical records, the framework takes into both pre-enhanced user profiles and users' historical records as input. The records are embedded, purified, and fused with the backbone model; and after the user profiles are embedded, a multi-interest extraction module is utilized with dual two-layer MLPs and a CapsNet to project the user profile information into the historical records and cluster the users' interests. Finally, after initial recommendation results are generated, we use a simulated user agent to refine the recommendations to get a more precise outcome. Details of the multi-interest extraction and the simulated user are shown at the bottom of the figure.

B. User Modeling

User modeling is another hot spot in the recommendation field. Unlike item representation learning which solely learns item embedding, the user modeling learns user interests in a user-oriented manner. To achieve this that, various approaches have been proposed to enhance user modeling. MHCN [30] uses a multi-channel convolutional network to encode hypergraphs in social recommendation. A next-item recommendation framework [31] is proposed to utilize sequential hypergraphs for dynamic user modeling. LSTPM [32] combines long-term interests with short-term interests to comprehensively model users' interests and yields significant improvements over the state-of-the-art methods. Wu et al. [25] incorporates users' social relations to enhance user modeling. Although these methods attempt to enhance user representation, these enhancements are not enough to mine users' interests precisely. Inspired by the UMI model [1], we integrate historical records with enriched user profiles.

C. Language Model for Recommendation

From the very beginning, language models (LMs) have benefited the recommendation area in many ways. At first attempts, some works [33], [34] utilize traditional IDs to represent users or items. Thereby, they disregard the semantic understanding capabilities of LMs, thus having little boost in performance. M6Rec [35] and UniSRec [36] incorporate the natural language information as part of the users/items modeling, achieving significant results. The models above are limited to small models, while TallRec [26] and NIR [27] apply LLMs in recommendation scenes. TallRec [26] finetunes a pre-trained LLM with recommendation data to obtain a specialized LLM for recommendation, and NIR [27] proposes a zero-shot next-item recommendation prompting strategy to direct LLMs to make next-item recommendations. However, these works directly use LLM as a recommender system and ignore the fact that the LLM is not capable of handling collaborative signals, which are crucial to a recommender.

III. Method

Different from most recommender systems, we integrate users' historical records with user profiles, which come from both the original dataset and the mining results. Users' historical records contain a list of items the users have interacted with and their features include ItemID, visual features, textual features, timestamps, etc. An overview of our method is shown in Fig. 2. First, a backbone model MCLN [4] is employed to extract and fuse multi-modal features and model high-order item-item relations and uninterested preferences. Then, we use a user profile regulation module to re-weight the items, followed by a multi-interest extraction module to cluster these items into disentangled interest vectors. Finally, a simulated user agent is utilized to achieve real-time feedback to refine the recommendations.

A. Problem Formulation

Let $\mathcal{U}$ denote a set of users, and $\mathcal{I}$ denote a set of items (known as item pool). For a given user u, $\mathcal{I}_u$ is his/her interacted items (x_1, x_2, x_3, ..., x_t), where x_i is the i-th item interacted by the user and t is the historical behavior length. $\mathcal{P}_u$ is his/her user profile. The ultimate goal is to recommend items that align with the user's interests. Modern recommender systems usually adopt a two-stage approach, the retrieval stage and the ranking stage. The retrieval stage is to retrieve top-N candidate items from the item pool while the ranking stage is designed to rank the items based on specific criteria.

The task for the retrieval stage in recommender systems is to retrieve a subset of items that are related to the user interests from $\mathcal{I}$. Generally, the task in the retrieval stage with multi-interest extraction can be formulated as follows:

$$R_{u,t} = \max_{1 \le k \le K} (\boldsymbol{i}_u^k)^\top \mathbf{e}_t \quad (1)$$

where $\boldsymbol{i}_u^k$ denotes k-th interest vector for the user u and $\mathbf{e}_t$ denotes the representation vector for target item x_t. The relevance score between the users' interest vectors and the target item is denoted as $R_{u,t}$.

Our paper mainly focuses on the retrieval stage by improving the modeling of user-side information.

B. User Profile Integration

Previous models fail to realize the importance of integrating user profiles even if they have such data. In this part, we not only integrate user profile information into our framework but also enhance it with data mining.

User Profile Enhancement. In this part, we first enhance the user profile because, in reality, the direct data source only contains users' static demographic features like gender, age, occupation, location, etc, and we argue that such data is not enough to precisely model user information.

We enhance users' profiles with data mining. For economic status, we combine geographical and occupational information with external knowledge about the relations between the two features to label the users' income levels. For acceptance of new items and behavioral features, we analyze users' interaction history and categorize them into pre-defined groups.

The above process can be formulated as follows:

$$\mathbf{e}_u^e = f(\mathbf{e}_u^g + \mathbf{e}_u^o + k_{ext}) \tag{2}$$

$$\mathbf{e}_u^b = g(d_h) \tag{3}$$

where $\mathbf{e}_u^e$, $\mathbf{e}_u^g$, $\mathbf{e}_u^o$, and $\mathbf{e}_u^b$ are user's economic status, geographical information, occupations, and behavioral features respectively. k_{ext} denotes external knowledge and d_h denotes users' interaction history. $f(\cdot)$ and $g(\cdot)$ are labeling and categorization functions.

User Profile Injection. Normally, historical items can represent user interests to some extent. However, the user profile can inherently regulate the users' interests and identify the most influential part of users' diverse interests, so that the real interests can be extracted precisely. Given the embedding vector e_i of the *i*th historical item x_i and the user profile e_u, the regulation of the user profile onto items can be formulated as follows:

$$a_i = \sigma(\mathbf{W}_2\phi(\mathbf{W}_1[\mathbf{e}_i, \mathbf{e}_u] + \boldsymbol{b}_1) + b_2), \tag{4}$$

where $\sigma(\cdot)$ is sigmoid function, $\phi(\cdot)$ is ReLu function, $\mathbf{W}_1$, $\mathbf{W}_2$, $\boldsymbol{b}_1$ and b_2 are all trainable parameters.

C. Multi-Modality-Interest Extraction

Capsule Network has been proven to be effective in interest clustering in recommender systems [1], [2], [7], [10]. In CapsNet, a capsule is a cluster where input vectors related to the same category are aggregated together. Specifically, we adapt CapsNet to multi-modality scenes to generate multiple interests for the user. Given the the item representations $\boldsymbol{I} = [\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_m]$, the jth capsule calculates the interest vector $\boldsymbol{h}_j$ as follows:

$$\boldsymbol{h}_j = \sum_{i=1}^{m} \boldsymbol{c}_{ij}\mathbf{W}_j a_i \mathbf{e}_i, \tag{5}$$

where $\mathbf{W}_j$ is the transformation matrix, and $\boldsymbol{c}_{ij}$ are the coupling coefficients, indicating the clustering probability for item $\mathbf{e}_i$ under the jth interest capsule. It can be formulated as follows:

$$\boldsymbol{c}_{ij} = \frac{\exp \boldsymbol{r}_{ij}}{\sum_{k=1}^{K} \exp \boldsymbol{r}_{ik}}, \tag{6}$$

where $\boldsymbol{r}_{ij}$ are the prior logits, defined as follows:

$$\boldsymbol{r}_{ij} = (\mathbf{W}_j \boldsymbol{e}_i)^{\mathsf{T}} \boldsymbol{s}_j, \tag{7}$$

where $\boldsymbol{s}_j$ is the "squashed" interest vector:

$$\boldsymbol{s}_j = \gamma(\boldsymbol{h}_j) = \frac{||\boldsymbol{h}_j||^2}{||\boldsymbol{h}_j||^2 + 1} \frac{\boldsymbol{h}_j}{||\boldsymbol{h}_j||^2} \tag{8}$$

where $\boldsymbol{h}_j$ is the total input of capsule j, and $\gamma(\cdot)$ is the "squash" function.

From Eq. 5 to Eq. 8, we can see that the calculation of the output $\boldsymbol{s}_j$ relies on itself. As a result, a dynamic routing process is designed to update its value iteratively. We first initialize $\boldsymbol{r}_{ij}$ with Gaussian distribution.

Given the above process, we have K interest capsules after iteratively updating values.

These interest capsules cluster users' interests iteratively, guiding us to explore the interest categories of the user.

To further intensify the regulation of user profiles, we use a two-layer ReLU-MLP structure to calculate the reference weights of different aspects of user profiles on the interest vector, formulated as follows:

$$\boldsymbol{x}_k = \sigma(\mathbf{W}_4\phi(\mathbf{W}_3[\mathbf{s}_k, \mathbf{e}_u] + \boldsymbol{b}_3) + b_4), \tag{9}$$

where $\sigma(\cdot)$ is sigmoid function, $\phi(\cdot)$ is ReLu function, $\mathbf{W}_3$, $\mathbf{W}_4$, $\boldsymbol{b}_3$ and b_4 are all trainable parameters.

With the reference weights $\boldsymbol{x}_k$, we can formulate the interest-related user profile as follows:

$$\boldsymbol{e}_u^n = \delta(\boldsymbol{d}_{n1}\boldsymbol{e}_u^{n1}, \boldsymbol{d}_{n2}\boldsymbol{e}_u^{n2}, ..., \boldsymbol{d}_{nN}\boldsymbol{e}_u^{nN}) \tag{10}$$

where $\boldsymbol{d}_{ni}$ and $\boldsymbol{e}_u^{ni}$ are the *i*th element of $\boldsymbol{x}_k$ and $\boldsymbol{e}_u^n$ respectively, δ denotes the concatenation operation.

Finally, we formulate the *k*th user interest $\boldsymbol{o}_u^k$ through a two-layer MLP with ReLU as follows:

$$\boldsymbol{o}_u^k = h([\boldsymbol{e}_u^k, \boldsymbol{x}_k]) \tag{11}$$

where $h(\cdot)$ is an MLP.

Here, we can see the interest vector is regulated with relevant user profile features.

D. Simulated User Agent

When serving, we use a pre-trained simulated user agent to mimic the user behaviors and give feedback to the recommender to refine the recommendations. Specifically, through practical attempts, we use ChatGPT as the agent by using prompts to instruct it because we found that it performs well in acting as a virtual role and ranking tasks, which perfectly fits our requirements. Here are a few prompt templates.

Profile Module. In the profile module, we will give a brief introduction about the target user with our user profile information. Before the system starts, we will give a general task description to the agent as follows: *You will now act as* *an agent to simulate a user on a platform, I will give you some brief information and then recommend a set of products from the item pool. Your task is to simulate the user's interests and accept or reject the recommendation result based on the interests.*

To give a brief introduction about the target user with our user profile information, we design the following prompt template:

Template 1: You are going to simulate a ¡Gender¿ user called ¡Name¿ (¡Age¿, lives in ¡Location¿, and works/studies as ¡Occupation¿). You ¡Behavior_Patterns¿, and love ¡Interests¿ movies.

Memory Module. In the memory module, we will give the historical records to the LLM and make it memorize these interactions with the following prompt:

Template 2: As ¡Name¿, you have watched (in sequence, followed by your rating) ¡Movie_Name¿ (¡Rating¿), ¡Movie_Name¿ (¡Rating¿), ¡Movie_Name¿ (¡Rating¿), ...

Feedback Module. In the feedback module, we will give initial recommendation results generated from our recommender system, and guide it to give pseudo comments as real-time feedback to refine the recommendations with the following prompt:

Template 3: Now the recommender wants to recommend ¡Candidate_Name¿, ¡Candidate_Name¿, ¡Candidate_Name¿, ¡Candidate_Name¿, ..., and your task is to rank them. Please reply at the end with "my ranking as ¡Name¿ would be ¡Candidate_name¿, ¡Candidate_name¿, ¡Candidate_name¿ ..." when ranking.

Template 4: Now the recommender wants to recommend ¡Candidate_Name¿, ¡Candidate_Name¿, ¡Candidate_Name¿, ¡Candidate_Name¿, ..., and your task is to decide whether or not you will accept the recommendation results item by item. Please be notified that the recommender might make mistakes, so do not echo the recommender. Please reply at the end with "I would accept these recommendations: ¡Candidate_name¿, ¡Candidate_name¿, ...", here ¡Candidate_name¿ is the item you'd like to accept.

Specifically, we adopt a "2x" strategy, which means if we want to recommend top-N items, we will give initial recommendation results with top-2N items and guide the simulated user to rank them. Then the top-N of ranked items will be the final recommendations and the rest are filtered out. This is because we find that the LLM is good at list-wise tasks and we will later study how prompt designs affect the performance of the simulated user agent.

E. Model Training and Optimization

With the multiple user interests $\boldsymbol{O}_u$ and the target item representation $\boldsymbol{e}_t$, the probability that user u will interact with the target item t can be calculated as follows:

$$p_u^t = \boldsymbol{O}_u^{\top} \boldsymbol{e}_t \tag{12}$$

The model training is to maximize this probability score for each positive target item in the training set against the rest negative ones, and the loss function can be formulated as follows:

$$\mathcal{L}_{all} = \sum_{u,t} -\log p_u^t + \lambda ||\Theta||^2 \tag{13}$$

where $||\Theta||^2$ is the $\mathcal{L}_2$ norm of all model parameters, and λ denotes the hyperparameter.

For negative data sampling, we adopt an in-batch neg-sampling strategy instead of the commonly used random sampling. Due to the existence of exposure bias, the "unseen" items are not necessarily items that the user does not actually like. In this case, the random sampling strategy has a high chance of selecting an item that is in fact a positive sample as a negative sample, even though the user has not seen it. For in-batch neg-sampling, in a mini-batch, we treat other users' positive samples as the target user's negative samples and significantly reduce the probability of mistakenly selecting negative samples by narrowing down the selection range.

IV. Experiments

A. Experimental Settings

Datasets. Three public datasets are used for model evaluation, including Movielens [37], Beauty [38] and Taobao [39]. Movielens [37] is a widely used and acknowledged dataset with multiple versions that contains users' ratings (integers ranging from 1 to 5) on movies. Here we choose its 1M version, which contains the user profile information like age, gender, occupation, etc. For visual and textual features, we crawl on the Internet to search for each movie's poster and introduction. Amazon-Beauty is a subset of the Amazon Product Dataset [38]. The Amazon Product Dataset is a large dataset collection containing product metadata (including textual descriptions and images), users' ratings, and reviews. Taobao [39] provides product data with visual content only and users' purchase history.

Data Preparations. For data preparations, in line with standard procedures, we divide these three datasets by 8:1:1 as train, test, and validation sets respectively. For dataset splitting, due to the time sensitivity of our task, we first adopt a temporal ordering strategy to split interacted items on a sampled user. Then, we use a ratio-based splitting following common practice, instead of the leave-one-out method.

To reduce noise and ensure the quality of the datasets, we adopt a 5-core strategy that filters out users with less than 5 interactions, and treat ratings equal to or over 3 as positive samples and ratings less than 3 as negative samples. Table I summarizes detailed statistics of the three datasets after preprocessing.

It should be noted that all sensitive data has been desensitized to avoid privacy concerns during training; when serving, the data collector is separate from the recommendation model, so that the process is user-identifier-free.

Evaluation Metrics. We use two commonly used metrics for evaluation: **HitRate@N** and **NDCG@N**.

$$\text{HitRate@}N = \frac{1}{|\mathbf{U}|} \sum_{u \in \mathbf{U}} \delta(|I_{u,N} \cap I_u| > 0), \tag{14}$$

TABLE I
STATISTICS OF DATASETS AFTER PREPROCESSING.

Dataset	#Users	#Items	#Interactions
Movielens	6,040	3,416	999,611
Beauty	15,576	8,678	139,318
Taobao	12,539	8,735	83,648

where $\delta(\cdot)$ is the indicator function, $I_{u,N}$ is the set of top-N recommended items for user u, and I_u is the set of testing items for user u.

$$\mathrm{NDCG@}N = \frac{\mathrm{DCG@}N}{\mathrm{IDCG@}N} \tag{15}$$

where $\mathrm{DCG@}N$ is the discounted cumulative gain that considers the positions of correct recommended items and $\mathrm{IDCG@}N$ is the ideal discounted cumulative gain or the maximum possible value of $\mathrm{DCG@}N$.

Baselines. To demonstrate the effectiveness of UiM, we compare the proposed UiM with the following state-of-the-art recommendation methods.

BPRMF [5]: BPRMF is a classical model optimized with BPR loss. It maps user and item representations as latent vectors based on user-item interactions directly.

SVD++ [6]: This model incorporates the information of the user's historically interacted neighbors into the user embedding.

ComiRec [2]: This is a novel multi-interest recommendation model with a single modality. We use the ComiRec-DR version based on a dynamic routing mechanism.

UMI [1]: This is a state-of-the-art user-aware multi-interest learning model for candidate retrieval that fuses user profiles and a capsule network to generate recommendation lists.

VBPR [28]: This model employs pre-trained convolution neural networks to extract visual features of the items and integrate them into the item embeddings.

MMGCN [21]: This model applies GCNs for embedding propagation on interaction graphs with different modality data to capture user interests in different modalities.

MMGCL [3]: This model introduces a negative sampling technique that learns the correlation between modalities and disentangles the users' tastes in different modalities.

MCLN [4]: This is a multi-modal recommendation model that identifies and eliminates the user preference-irrelevant portion inside user-interacted multimodal content with counterfactual inference of causal theory.

All these baselines can be divided into three categories: (1) ID-based recommendation models (i.e., BPRMF, SVD++); (2) Multi-interest-based models (i.e., ComiRec, UMI); (3) Multi-modal models (i.e., VBPR, MMGCN, MCLN).

1) Hyperparameter Settings: For a fair comparison, all methods are implemented with Pytorch 2.0 in Python 3.9.16 and learned with Adam optimizer [40].

We conduct our experiments on a single Linux server with 2 Intel(R) Xeon(R) CPU Gold 6132 @2.60GHz, 256GB RAM, and 4 NVIDIA GeForce RTX 3090 (24GB each). The learning rate and mini-batch size are set to 1×10^{-3} and 128 for the Movielens dataset, and 1×10^{-3} and 256 for the rest two datasets. The number of negative samples is 5 in the training stage for all three datasets. We tuned the parameters of all methods over the validation and set the embedding size as 512 and 1024 for the Movielens dataset and the rest two datasets respectively.

B. Overall Performance

Table II summarizes the overall performance of all models. Here, we make the following observations.

First, the multi-modal models (VBPR, MMGCN, MMGCL, MCLN) are greatly superior to the ID-based models (BPRMF, SVD++). This demonstrates the effectiveness of integrating multi-modal information into embedding generation to assist in modeling user interests. Among all multi-modal baseline models, MCLN performs best on all datasets, which can capture high-order collaborative signals with GCN and remove the irrelevant parts of users' interacted items, purifying the representations.

Second, the multi-interest baseline models (UMI, ComiRec) are significantly stronger than BPRMF and SVD++ on these three datasets, which verifies the effectiveness of capturing multiple disentangled interests to enhance the expressiveness of the user embedding. In all cases, UMI outperforms ComiRec. A possible reason is that the UMI not only utilizes multi-interest extraction with ID embeddings but also integrates users' static demographic attributes.

Third, the multi-modal models outperform ComiRec, suggesting that visual features are of great importance in visually enhanced fields like movie recommendation, but fall short of UMI. We infer that the direct usage of user-side information greatly improves the modeling of user interests and surpasses the enhancement of item representation, which is used to indirectly model user interests.

Our proposed UiM method outperforms state-of-the-art baseline models in all settings, showing strong effectiveness in recommendation accuracy. Compared with other multi-interest and multi-modal models, our method utilizes some design choices like user information enhancement and incorporation, multi-modal multi-interest mining, and a simulated user agent, leading to better performance. Specifically, our UiM method outperforms the MCLN model in all experiment settings, which demonstrates that integrating user-side information shows potential in multi-modal recommendation scenes. Further, our UiM model surpasses the UMI model which utilizes static user profiles for single-modal recommendation in all settings in both metrics, showing the value of user profile enhancement and multi-modal multi-interest design.

C. Ablation Study

Study on the impacts of design choices. We conduct an ablation study on the Beauty Dataset for each design choice in our model to validate their effectiveness. Specifically, these factors include enhanced user profile (Profile), multi-interest extraction (Multi), and simulated feedback (S-F). Table III reports the performance of these variants and the full model on the Beauty dataset. Here, we can make the following observations.

TABLE II
THE OVERALL PERFORMANCE OF DIFFERENT METHODS ON THE THREE DATASETS. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDFACE. THE RESULTS ARE PERCENTAGE NUMBERS WITH "%" OMITTED.

Methods	Movielens				Beauty				Taobao			
	metrics@10		metrics@50		metrics@10		metrics@50		metrics@10		metrics@50	
	HitRate	NDCG	HitRate	NDCG	HitRate	NDCG	HitRate	NDCG	HitRate	NDCG	HitRate	NDCG
BPRMF [5]	16.39	13.69	22.68	19.47	51.73	36.34	69.39	42.05	40.49	27.33	42.45	27.25
SVD++ [6]	16.77	14.02	23.12	20.61	55.20	38.95	69.73	42.95	42.93	28.19	43.72	28.31
ComiRec [2]	18.92	14.88	24.30	21.75	56.54	39.22	71.52	43.41	43.55	29.30	52.42	31.37
UMI [1]	19.36	16.07	25.91	21.94	58.36	41.02	73.49	45.17	47.02	30.24	60.35	33.28
VBPR [28]	18.98	16.55	25.82	23.48	56.70	39.73	72.36	45.52	43.64	29.28	59.82	33.02
MMGCN [21]	20.04	16.92	27.20	25.75	60.67	40.81	74.09	47.65	46.95	30.47	60.67	33.24
MMGCL [3]	21.33	17.45	29.45	27.59	64.78	44.89	75.21	49.38	48.33	32.56	63.29	34.45
MCLN [4] (baseline)	23.11	20.25	34.10	28.87	66.89	47.10	76.27	52.02	51.03	33.93	74.26	35.65
UiM (Ours)	**25.32**	**22.08**	**35.27**	**30.32**	**71.66**	**50.15**	**79.30**	**54.12**	**53.83**	**36.11**	**75.35**	**38.22**

TABLE III
THE ABLATION STUDY OF UIM ON THE BEAUTY DATASET. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDFACE. THE RESULTS ARE PERCENTAGE NUMBERS WITH "%" OMITTED.

Module			Beauty	
Profile	Multi	S-F	HR@10	NDCG@10
○	○	○	66.89	47.10
●	○	○	67.02	47.93
○	●	○	68.39	48.04
○	○	●	68.95	48.07
●	●	○	69.72	48.82
●	○	●	71.03	49.92
○	●	●	70.24	49.25
●	●	●	**71.66**	**50.15**

TABLE IV
THE ABLATION STUDY OF PROMPT DESIGNS ON THE MOVIELENS DATASET. "S-U" MEANS SIMULATED USER. THE BEST RESULTS ARE HIGHLIGHTED IN BOLDFACE. THE RESULTS ARE PERCENTAGE NUMBERS WITH "%" OMITTED.

Prompt Design	HR@10	NDCG@10	HR@20	NDCG@20
w/o S-U	24.19	20.98	29.77	27.89
S-U w/ Prompt a	**25.32**	**22.08**	**30.79**	**28.93**
S-U w/ Prompt b	24.72	21.49	30.32	28.02

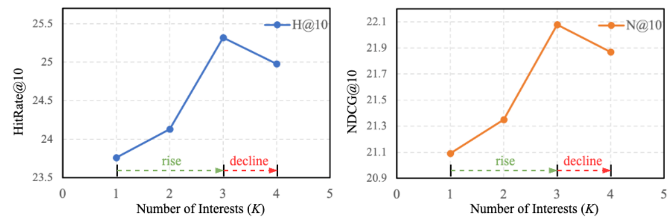

Fig. 3. The performance of different K values on the Movielens dataset. The left subfigure plots the HitRate@10 metric; and the right subfigure shows the NDCG@10 metric.

Firstly, adding the multi-modality-interest extraction module alone can witness a performance jump over the baseline model, suggesting the effectiveness of generating disentangled interest features. Secondly, performance experiences a significant rise after the simulated user agent is implemented to achieve real-time feedback, showing the strong ability to refine recommendations with an LLM.

It should be noted that, in the comparison of single design choices, adding the multi-modal multi-interest learning alone is more accurate than adding the user profile information alone. But when a simulated user has been added, the reverse is true. A possible explanation is that the simulated user agent has a broad grasp of external knowledge and can more effectively capture user interests by combining such knowledge with user profile information.

Study on the impact of prompt designs. To ensure the generalization ability of our framework, we conduct an ablation study on the Movielens dataset to study the impact of different prompt designs and report the performance in table IV. Specifically, we use two prompt designs (denoted as *Template a* and *Template b*) in the Feedback Module:

a) *Template a*: Now the recommender wants to recommend ¡Candidate_Name¿, ¡Candidate_Name¿, ¡Candidate_Name¿, ¡Candidate_Name¿, . . . , and your task is to decide whether or not you will accept the recommendation results.

b) *Template b*: Now the recommender wants to recommend ¡Candidate_Name¿, ¡Candidate_Name¿, ¡Candidate_Name¿, ¡Candidate_Name¿, . . . , and your task is to decide whether or not you will accept the recommendation results item by item and rank them. Please be notified that the recommender might make mistakes, so do not echo the recommender. Please reply at the end with "I would accept these recommendations: ¡Candidate_name¿, ¡Candidate_name¿, ...", here ¡Candidate_name¿ is the item you'd like to accept and "my ranking as ¡Name¿ would be ¡Candidate_name¿ ¿ ¡Candidate_name¿ ¿ ¡Candidate_name¿ ..." when ranking.

As explained in section 3.4, the simulated user agent does well in list-wise tasks as a result of Transformer-based architecture. From table IV, we can see that the simulated user with *Template a* has a little performance gain compared with the simulated user not used setting. The reason is that *Template a* instructs the simulated user to give overall feedback on the recommendation results, and the agent will tend to accept the results if only part of the recommendation results are correct, although the remaining parts are incorrect, resulting in its correction function not being fully exerted. Also, *Template b* has better performance compared with *Template a* and verifies the above finding that the agent does well in list-wise tasks.

Study on the impact of K Value. The number of interests K in UiM controls the diversity of user interests. Fig. 3

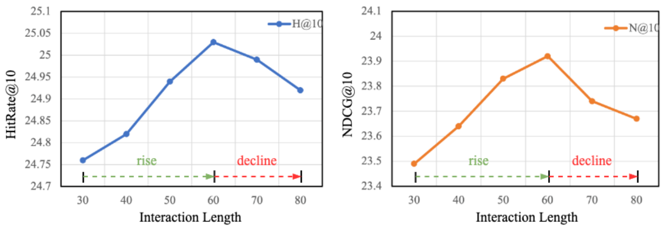

Fig. 4. The performance of different lengths of interactions on the Movielens dataset. The left subfigure plots the HitRate@10 metric; and the right subfigure shows the NDCG@10 metric.

plots the performance changes of varying K values for the Movielens dataset. We can observe that a single interest representation (i.e., $K = 1$) achieves the worst performance across the two metrics, which is the common practice in most current methods. The optimal K value is 3 for the Movielens dataset.

We made some assumptions about the decline when K increases from 3 to 4. We analyzed the dataset and found that most users do not have a significant number of interest categories. So when the number of interest capsules continues to grow after reaching its optimal value, the performance will experience a decrease instead of the expected increase.

Study on the impact of Interaction Length. The length of interaction history in UiM is highly related to recommendations. Fig. 4 plots the performance changes of varying length values for the Movielens dataset. We can observe that a shorter length of interactions achieves less performance across the two metrics, which is consistent with common sense. However, when a much longer length of interactions comes into the recommender system, the performance peak is not shown as expected across the two metrics. Noise might be the culprit that pollutes the representation of user interests when interaction history becomes longer.

Study on cold-start recommendations. We perform additional experiments in cold-start scenarios by manually truncating the user history data to a length of 5, 10, and 30 in the Movielens dataset. The performances are reported in Table V.

From the table, the following observations can be made:

First, across all length settings, performances for metrics@1 are worse than metrics@5. This is reasonable because when calculating HitRate, the denominator is the number of interacted items. Given a specific length of history data, when N arises, the chance of having more right items arises, making the numerator greater while the denominator remains unchanged.

Second, across all metrics, performances under the length of 30, which corresponds to the warm-start scenario, are better than those under 5 and 10. The cold-start challenge has affected numerous recommender systems. Many deployed systems [41]–[43] utilize techniques like meta-learning to solve this issue.

Third, in cold-start settings, the performance under the length of 10 is slightly better than that of 5, which verified the fact that a shorter interaction history leads to worse recommendation accuracy in cold-start scenes.

Specifically, our system, with the incorporation of user-side information, multi-modal multi-interest extraction, and the simulated user agent, reduces the reliance on historical ID embedding. So it can be seen that the performance degradation of our model is not that much in cold-start settings compared with the warm-start setting. The zero-shot ability of the simulated user agent component also ensures the performance of our model in cold-start scenarios.

TABLE V
THE PERFORMANCE IN COLD-START SCENARIOS IN THE MOVIELENS DATASET. THE RESULTS ARE PERCENTAGE NUMBERS WITH ”%” OMITTED.

Truncated length	HR@1	NDCG@1	HR@5	NDCG@5
5	13.24	12.88	15.09	14.83
10	14.96	13.67	16.41	16.10
30 (warm)	16.53	15.49	19.35	19.07

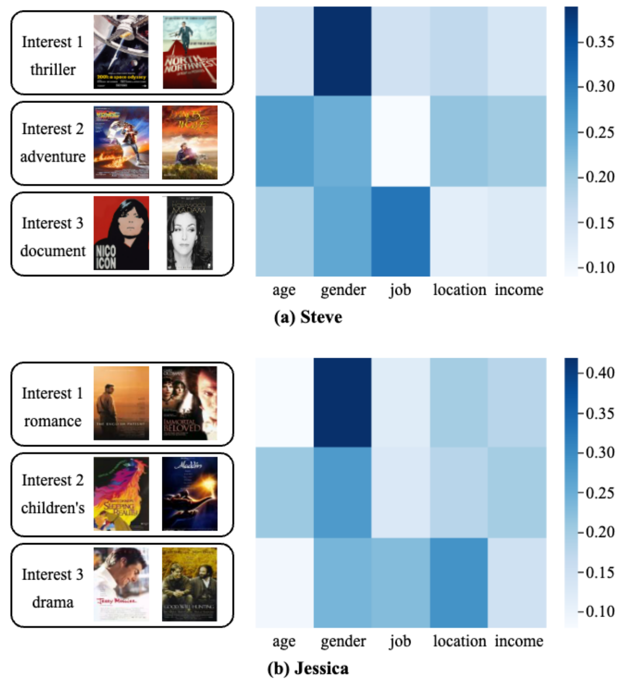

Fig. 5. Steve’s and Jessica’s user profiles and the impacts on their interests. On the left are interest categories; on the right are the reference weights the interest categories have on the profile features. (Best viewed in color)

D. Case Study

In this part, we randomly select a male user (namely Steve) and a female user (namely Jessica) from the Movielens dataset and perform several case studies to intuitively display our model performance. Steve’s historical records can be roughly categorized into six groups: thriller, adventure, document, animation, sci-fi, and action; while Jessica’s historical records can be divided into five groups: romance, children’s, drama, musical, and mystery. Due to limited space, we only demonstrate three main interests for them.

Study on the impact of user profile. In this experiment, we use Steve’s and Jessica’s data and display the reference weights provided by equation 9 for the user profile features. In addition

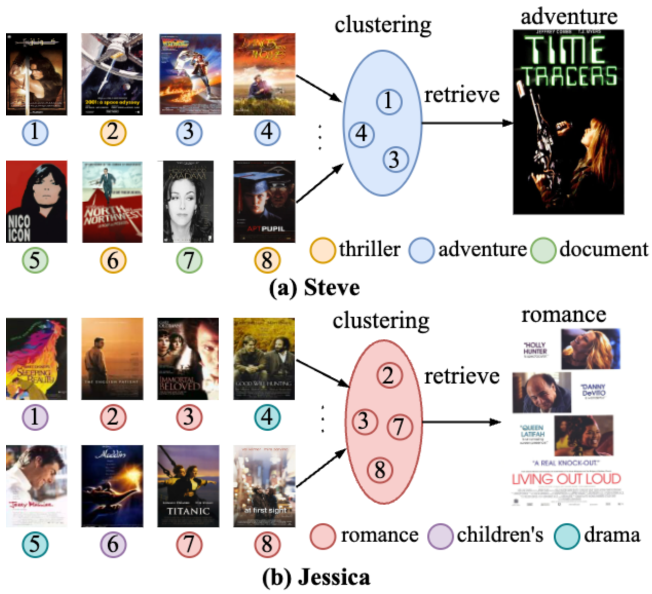

Fig. 6. Visualization of multi-interest extraction on Steve and Jessica. On the left are the historical items in time order; on the right are examples of recommendations generated by our framework with multi-interest extraction.

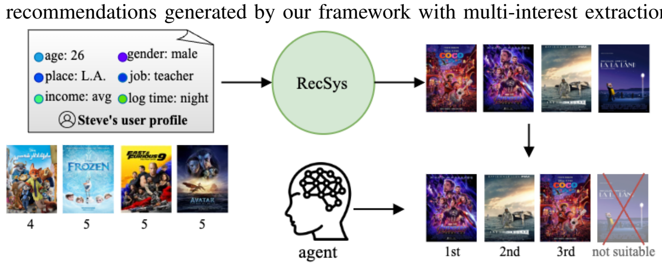

Fig. 7. Visualization of the workflow of the simulated user agent. The recommender system takes the user's profile and historical items with ratings, generates initial recommendation results, and passes the initial results to the simulated user. Then, the virtual user that simulates the real user gives real-time feedback to refine the recommendations.

to the reference weights, we display the clicked movies relevant to each interest. Fig. 5 shows Steve's and Jessica's user profiles and their impacts on the user's interests, respectively. From the figure, we can see that Steve is interested in thriller movies and that Jessica shows a particular interest in romance movies, which differs from Steve's interests. A possible reason is that male users tend to like movies with horrifying elements while female users are more likely to watch movies with romance, drama, amusement, etc, and the darker colors of gender and age on the thriller/romance interests in Fig. 5 give us an intuitive glimpse into how user profile features affect users' interests.

Study on the impact of multi-interest extraction. Later, we perform multi-interest extraction experiments on Steve and Jessica, and the results are shown in Fig. 6. From Fig. 6, we can see that Steve's and Jessica's interests are divided into different categories, which later can be used to generate candidate items, as shown in the right part of Fig. 6.

Study on the impact of the simulated user agent. Also, we perform experiments with the simulated user agent on Steve and visualize the workflow in Fig. 7. In this experiment, we use the rest part of Steve's interaction history as a toy example and show how the simulated user agent works as a real-time feedback source. In the example, we provide the agent with Steve's user profile information using *Template 1* in the Profile Module. Later, we utilize *Template 2* in the Memory Module to let the agent memorize the interaction history. Finally, *Template 3* in the Feedback Module is utilized to get pseudo comments as real-time feedback to refine the recommendations.

Simulated User: As Steve, based on my movie preferences and interests, I would assess each recommendation separately:

a. *COCO*: I would likely accept this recommendation, as I have previously enjoyed animated films like Zootopia and Frozen.

b. *Avengers*: I would definitely accept this recommendation, as I love action movies and have given a high rating to Fast & Furious 9.

c. *La La Land*: Considering my preferences for animation, sci-fi, and action movies, La La Land, being a musical and romantic drama, may not align closely with my usual choices. Therefore, I might be hesitant to accept this recommendation.

d. *Interstellar*: Given my interest in sci-fi, I would also accept this recommendation, as it aligns with my preferences.

In summary, as Steve, I would accept these recommendations: *COCO*, *the Avengers*, and *Interstellar*. They cover a range of genres that I enjoy, including animation, action, and sci-fi, and my ranking as Steve would be *Avengers ¿ Interstellar ¿ COCO*.

V. Discussion

In this paper, we examine the integration of user-side information and propose a novel user-insight multi-modal framework for recommendation. Specifically, the framework is capable of dealing with user profile information by enhancement and fusion with historical records. Also, a simulated user is utilized to refine the recommendations. However, our method still has certain limitations: 1. The effectiveness of the simulated user agent depends on the LLM, which may not always be accessible; 2. Our method integrates user-side information with item representations that already fuse multi-modal features. However, users' tastes in different modalities are not the same. In the future, we will explore user profile integration on different modalities and try to expand our work into more fields like music recommendation.

VI. Conclusion

In this paper, we propose a novel user-insight multi-modal recommendation framework, enabling more precise candidate retrieval in multi-modal recommenders. We enhance user profiles and integrate them into the framework, achieving more accurate interest extraction. We also tailor Capsule Network to better exploit users' diverse interests in the field of MMRec. Specifically, we introduce a simulated user agent to achieve real-time feedback. Experiments through three public real-world datasets confirm the superiority of our proposed method.

References

[1] Z. Chai, Z. Chen, C. Li, *et al.*, "User-aware multi-interest learning for candidate matching in recommenders," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '22, New York, NY, USA: Association for Computing Machinery, 2022, 1326–1335. DOI: 10.1145/3477495.3532073. [Online]. Available: https://doi.org/10.1145/3477495.3532073.

[2] Y. Cen, J. Zhang, X. Zou, C. Zhou, H. Yang, and J. Tang, "Controllable multi-interest framework for recommendation," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '20, New York, NY, USA: Association for Computing Machinery, 2020, 2942–2951. DOI: 10.1145/3394486.3403344. [Online]. Available: https://doi.org/10.1145/3394486.3403344.

[3] Z. Yi, X. Wang, I. Ounis, and C. Macdonald, "Multi-modal graph contrastive learning for micro-video recommendation," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '22, Madrid, Spain: Association for Computing Machinery, 2022, 1807–1811, ISBN: 9781450387323. DOI: 10.1145/3477495.3532027. [Online]. Available: https://doi.org/10.1145/3477495.3532027.

[4] S. Li, D. Guo, K. Liu, R. Hong, and F. Xue, "Multi-modal counterfactual learning network for multimedia-based recommendation," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '23, Taipei, Taiwan: Association for Computing Machinery, 2023, 1539–1548, ISBN: 9781450394086. DOI: 10.1145/3539618.3591739. [Online]. Available: https://doi.org/10.1145/3539618.3591739.

[5] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, *Bpr: Bayesian personalized ranking from implicit feedback*, 2012. arXiv: 1205.2618 [cs.IR].

[6] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '08, Las Vegas, Nevada, USA: Association for Computing Machinery, 2008, 426–434, ISBN: 9781605581934. DOI: 10.1145/1401890.1401944. [Online]. Available: https://doi.org/10.1145/1401890.1401944.

[7] C. Li, Z. Liu, M. Wu, *et al.*, "Multi-interest network with dynamic routing for recommendation at tmall," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, ser. CIKM '19, New York, NY, USA: Association for Computing Machinery, 2019, 2615–2623. DOI: 10.1145/3357384.3357814. [Online]. Available: https://doi.org/10.1145/3357384.3357814.

[8] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR'19, Paris, France: Association for Computing Machinery, 2019, 165–174, ISBN: 9781450361729. DOI: 10.1145/3331184.3331267. [Online]. Available: https://doi.org/10.1145/3331184.3331267.

[9] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "Lightgcn: Simplifying and powering graph convolution network for recommendation," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '20, Virtual Event, China: Association for Computing Machinery, 2020, 639–648, ISBN: 9781450380164. DOI: 10.1145/3397271.3401063. [Online]. Available: https://doi.org/10.1145/3397271.3401063.

[10] Y. Tian, J. Chang, Y. Niu, Y. Song, and C. Li, "When multi-level meets multi-interest: A multi-grained neural model for sequential recommendation," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '22, Madrid, Spain: Association for Computing Machinery, 2022, 1632–1641, ISBN: 9781450387323. DOI: 10.1145/3477495.3532081. [Online]. Available: https://doi.org/10.1145/3477495.3532081.

[11] X. Wang, X. He, Y. Cao, M. Liu, and T.-S. Chua, "Kgat: Knowledge graph attention network for recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '19, New York, NY, USA: Association for Computing Machinery, 2019, 950–958. DOI: 10.1145/3292500.3330989. [Online]. Available: https://doi.org/10.1145/3292500.3330989.

[12] Z. Yuan, F. Yuan, Y. Song, *et al.*, "Where to go next for recommender systems? id- vs. modality-based recommender models revisited," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '23, Taipei, Taiwan: Association for Computing Machinery, 2023, 2639–2649, ISBN: 9781450394086. DOI: 10.1145/3539618.3591932. [Online]. Available: https://doi.org/10.1145/3539618.3591932.

[13] H. Liu, W. Wang, H. Xu, Q. Peng, and P. Jiao, "Neural unified review recommendation with cross attention," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '20, Virtual Event, China: Association for Computing Machinery, 2020, 1789–1792, ISBN: 9781450380164. DOI: 10.1145/3397271.3401249. [Online]. Available: https://doi.org/10.1145/3397271.3401249.

[14] C. Chen, M. Zhang, Y. Liu, and S. Ma, "Neural attentional rating regression with review-level explanations," in *Proceedings of the 2018 World Wide Web Conference*, ser. WWW '18, Lyon, France: International

World Wide Web Conferences Steering Committee, 2018, 1583–1592, ISBN: 9781450356398. DOI: 10.1145/3178876.3186070. [Online]. Available: https://doi.org/10.1145/3178876.3186070.

[15] D. Liu, J. Li, B. Du, J. Chang, and R. Gao, "Daml: Dual attention mutual learning between ratings and reviews for item recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '19, Anchorage, AK, USA: Association for Computing Machinery, 2019, 344–352, ISBN: 9781450362016. DOI: 10.1145/3292500.3330906. [Online]. Available: https://doi.org/10.1145/3292500.3330906.

[16] Y. Lu, R. Dong, and B. Smyth, "Coevolutionary recommendation model: Mutual learning between ratings and reviews," in *Proceedings of the 2018 World Wide Web Conference*, ser. WWW '18, Lyon, France: International World Wide Web Conferences Steering Committee, 2018, 773–782, ISBN: 9781450356398. DOI: 10.1145/3178876.3186158. [Online]. Available: https://doi.org/10.1145/3178876.3186158.

[17] C. Huang, S. Wang, X. Wang, and L. Yao, "Dual contrastive transformer for hierarchical preference modeling in sequential recommendation," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '23, Taipei, Taiwan: Association for Computing Machinery, 2023, 99–109, ISBN: 9781450394086. DOI: 10.1145/3539618.3591672. [Online]. Available: https://doi.org/10.1145/3539618.3591672.

[18] Q. Liu, S. Wu, and L. Wang, "Deepstyle: Learning user preferences for visual recommendation," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '17, Shinjuku, Tokyo, Japan: Association for Computing Machinery, 2017, 841–844, ISBN: 9781450350228. DOI: 10.1145/3077136.3080658. [Online]. Available: https://doi.org/10.1145/3077136.3080658.

[19] X. Chen, H. Chen, H. Xu, *et al.*, "Personalized fashion recommendation with visual explanations based on multimodal attention network: Towards visually explainable recommendation," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR'19, Paris, France: Association for Computing Machinery, 2019, 765–774, ISBN: 9781450361729. DOI: 10.1145/3331184.3331254. [Online]. Available: https://doi.org/10.1145/3331184.3331254.

[20] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, "Attentive collaborative filtering: Multimedia recommendation with item- and component-level attention," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '17, Shinjuku, Tokyo, Japan: Association for Computing Machinery, 2017, 335–344, ISBN: 9781450350228. DOI: 10.1145/3077136.3080797. [Online]. Available: https://doi.org/10.1145/3077136.3080797.

[21] Y. Wei, X. Wang, L. Nie, X. He, R. Hong, and T.-S. Chua, "Mmgcn: Multi-modal graph convolution network for personalized recommendation of micro-video," in *Proceedings of the 27th ACM International Conference on Multimedia*, ser. MM '19, Nice, France: Association for Computing Machinery, 2019, 1437–1445, ISBN: 9781450368896. DOI: 10.1145/3343031.3351034. [Online]. Available: https://doi.org/10.1145/3343031.3351034.

[22] J. Zhang, Y. Zhu, Q. Liu, S. Wu, S. Wang, and L. Wang, "Mining latent structures for multimedia recommendation," in *Proceedings of the 29th ACM International Conference on Multimedia*, ser. MM '21, Virtual Event, China: Association for Computing Machinery, 2021, 3872–3880, ISBN: 9781450386517. DOI: 10.1145/3474085.3475259. [Online]. Available: https://doi.org/10.1145/3474085.3475259.

[23] X. Zhou and C. Miao, "Disentangled graph variational auto-encoder for multimodal recommendation with interpretability," *IEEE Transactions on Multimedia*, pp. 1–13, 2024. DOI: 10.1109/TMM.2024.3369875.

[24] R. Xu, J. Wang, and Y. Li, "Heterogeneous-grained multi-modal graph network for outfit recommendation," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 8, no. 2, pp. 1788–1799, 2024. DOI: 10.1109/TETCI.2024.3358190.

[25] L. Wu, J. Li, P. Sun, R. Hong, Y. Ge, and M. Wang, "Diffnet++: A neural influence and interest diffusion network for social recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 10, pp. 4753–4766, 2022. DOI: 10.1109/TKDE.2020.3048414.

[26] K. Bao, J. Zhang, Y. Zhang, W. Wang, F. Feng, and X. He, "Tallrec: An effective and efficient tuning framework to align large language model with recommendation," in *Proceedings of the 17th ACM Conference on Recommender Systems*, ser. RecSys '23, Singapore, Singapore: Association for Computing Machinery, 2023, 1007–1014, ISBN: 9798400702419. DOI: 10.1145/3604915.3608857. [Online]. Available: https://doi.org/10.1145/3604915.3608857.

[27] L. Wang and E.-P. Lim, *Zero-shot next-item recommendation using large pretrained language models*, 2023. arXiv: 2304.03153 [cs.IR].

[28] R. He and J. McAuley, "Vbpr: Visual bayesian personalized ranking from implicit feedback," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30, no. 1, pp. 144–150, 2016. DOI: 10.1609/aaai.v30i1.9973. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/9973.

[29] C. Xu, Z. Guan, W. Zhao, *et al.*, "Recommendation by users' multimodal preferences for smart city applications," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4197–4205, 2021. DOI: 10.1109/TII.2020.3008923.

[30] J. Yu, H. Yin, J. Li, Q. Wang, N. Q. V. Hung, and X. Zhang, "Self-supervised multi-channel hypergraph convolutional network for social recommendation," in *Proceedings of the Web Conference 2021*, ser. WWW '21, Ljubljana, Slovenia: Association for Computing Machinery, 2021, 413–424, ISBN: 9781450383127. DOI: 10.1145/3442381.3449844. [Online]. Available: https://doi.org/10.1145/3442381.3449844.

[31] J. Wang, K. Ding, L. Hong, H. Liu, and J. Caverlee, "Next-item recommendation with sequential hypergraphs," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '20, Virtual Event, China: Association for Computing Machinery, 2020, 1101–1110, ISBN: 9781450380164. DOI: 10.1145/3397271.3401133. [Online]. Available: https://doi.org/10.1145/3397271.3401133.

[32] K. Sun, T. Qian, T. Chen, Y. Liang, Q. V. H. Nguyen, and H. Yin, "Where to go next: Modeling long- and short-term user preferences for point-of-interest recommendation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, pp. 214–221, 2020. DOI: 10.1609/aaai.v34i01.5353. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/5353.

[33] S. Geng, S. Liu, Z. Fu, Y. Ge, and Y. Zhang, "Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5)," in *Proceedings of the 16th ACM Conference on Recommender Systems*, ser. RecSys '22, Seattle, WA, USA: Association for Computing Machinery, 2022, 299–315, ISBN: 9781450392785. DOI: 10.1145/3523227.3546767. [Online]. Available: https://doi.org/10.1145/3523227.3546767.

[34] L. Li, Y. Zhang, and L. Chen, "Personalized prompt learning for explainable recommendation," *ACM Trans. Inf. Syst.*, vol. 41, no. 4, 2023, ISSN: 1046-8188. DOI: 10.1145/3580488. [Online]. Available: https://doi.org/10.1145/3580488.

[35] Z. Cui, J. Ma, C. Zhou, J. Zhou, and H. Yang, *M6-rec: Generative pretrained language models are open-ended recommender systems*, 2022. arXiv: 2205.08084 [cs.IR].

[36] Y. Hou, S. Mu, W. X. Zhao, Y. Li, B. Ding, and J.-R. Wen, "Towards universal sequence representation learning for recommender systems," in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD '22, Washington DC, USA: Association for Computing Machinery, 2022, 585–593, ISBN: 9781450393850. DOI: 10.1145/3534678.3539381. [Online]. Available: https://doi.org/10.1145/3534678.3539381.

[37] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, 2015, ISSN: 2160-6455. DOI: 10.1145/2827872. [Online]. Available: https://doi.org/10.1145/2827872.

[38] J. McAuley, C. Targett, Q. Shi, and A. van den Hengel, "Image-based recommendations on styles and substitutes," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '15, Santiago, Chile: Association for Computing Machinery, 2015, 43–52, ISBN: 9781450336215. DOI: 10.1145/2766462.2767755. [Online]. Available: https://doi.org/10.1145/2766462.2767755.

[39] Tianchi, *Fashion collocation data on taobao.com*, 2018. [Online]. Available: https://tianchi.aliyun.com/dataset/dataDetail?dataId=52.

[40] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, Jan. 2017. [Online]. Available: https://arxiv.org/abs/1412.6980.

[41] X. Pan, Y. Chen, C. Tian, *et al.*, "Multimodal meta-learning for cold-start sequential recommendation," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, ser. CIKM '22, Atlanta, GA, USA: Association for Computing Machinery, 2022, 3421–3430, ISBN: 9781450392365. DOI: 10.1145/3511808.3557101. [Online]. Available: https://doi.org/10.1145/3511808.3557101.

[42] H. Liu, H. Lin, X. Zhang, *et al.*, "Boosting meta-learning cold-start recommendation with graph neural network," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, ser. CIKM '23, Birmingham, United Kingdom: Association for Computing Machinery, 2023, 4105–4109, ISBN: 9798400701245. DOI: 10.1145/3583780.3615283. [Online]. Available: https://doi.org/10.1145/3583780.3615283.

[43] X. Huang, J. Sang, J. Yu, and C. Xu, "Learning to learn a cold-start sequential recommender," *ACM Trans. Inf. Syst.*, vol. 40, no. 2, 2022, ISSN: 1046-8188. DOI: 10.1145/3466753. [Online]. Available: https://doi.org/10.1145/3466753.

List of Figures